Multidimensional HLLC Riemann solver for unstructured meshes – with application to Euler and MHD flows

Balsara, Dinshaw S ; Dumbser, Michael ; Abgrall, Rémi

Abstract: The goal of this paper is to formulate genuinely multidimensional HLL and HLLC Riemann solvers for unstructured meshes by extending our prior papers on the same topic for logically rectangular meshes Balsara (2010, 2012) [4,5]. Such Riemann solvers operate at each vertex of a mesh and accept as an input the set of states that come together at that vertex. The mesh geometry around that vertex is also one of the inputs of the Riemann solver. The outputs are the resolved state and multidimensionally upwinded fluxes in both directions. A formulation which respects the detailed geometry of the unstructured mesh is presented. Closed-form expressions are provided for all the integrals, making it particularly easy to implement the present multidimensional Riemann solvers in existing numerical codes. While it is visually demonstrated for three states coming together at a vertex, our formulation is general enough to treat multiple states (or zones with arbitrary geometry) coming together at a vertex. The present formulation is very useful for two-dimensional and three-dimensional unstructured mesh calculations of conservation laws. It has been demonstrated to work with second to fourth order finite volume schemes on two-dimensional unstructured meshes. On general triangular grids an arbitrary number of states might come together at a vertex of the primal mesh, while for calculations on the dual mesh usually three states come together at a grid vertex. We apply the multidimensional Riemann solvers to hydrodynamics and magnetohydrodynamics (MHD) on unstructured meshes. The Riemann solver is shown to operate well for traditional second order accurate total variation diminishing (TVD) schemes as well as for weighted essentially non-oscillatory (WENO) schemes with ADER (Arbitrary DERivatives in space and time) time-stepping. Several stringent applications for compressible gasdynamics and magnetohydrodynamics are presented, showing that the method performs very well and reaches high order of accuracy in both space and time. The present multidimensional Riemann solver is cost-competitive with traditional, one-dimensional Riemann solvers. It offers the twin advantages of isotropic propagation of flow features and a larger CFL number.

DOI: https://doi.org/10.1016/j.jcp.2013.12.029

Originally published at:
Balsara, Dinshaw S; Dumbser, Michael; Abgrall, Rémi (2014). Multidimensional HLLC Riemann solver for unstructured meshes – with application to Euler and MHD flows. Journal of Computational Physics, 261:172-208.
DOI: https://doi.org/10.1016/j.jcp.2013.12.029

Multidimensional HLLC Riemann Solver for Unstructured Meshes – With Application to Euler and MHD Flows

By

Dinshaw S. Balsara[1,2] Michael Dumbser[3] and Remi Abgrall[4]

(dbalsara@nd.edu, michael.dumbser@iag.uni-stuttgart.de, remi.abgrall@inria.fr)

[1] Physics Department, University of Notre Dame, 225 Nieuwland Science Hall, Notre Dame, IN, 46556, USA

[2] Applied and Computational Mathematics and Statistics Department, University of Notre Dame, Hayes-Healey Hall, Notre Dame, IN, 46556, USA

[3] Laboratory of Applied Mathematics, University of Trento, Via Mesiano 77, I-38100 Trento, Italy

[4]INRIA Bordeaux, Sud-Ouest, Team BACCHUS, 351 Cours de la Libération, 33405 Talence cedex, France

Abstract

The goal of this paper is to formulate genuinely multidimensional HLL and HLLC Riemann solvers for unstructured meshes by extending our prior papers on the same topic for logically rectangular meshes (D.S. Balsara, *Multidimensional HLLE Riemann solver; Application to Euler and Magnetohydrodynamic Flows*, J. Comput. Phys., 229 (2010) 1970-1993; D.S. Balsara, *A two-dimensional HLLC Riemann solver for conservation laws: Application to Euler and MHD flows*, Journal of Computational Physics, 231 (2012) 7476-7503). Such Riemann solvers operate at each vertex of a mesh and accept as an input the set of states that come together at that vertex. The mesh geometry around that vertex is also one of the inputs of the Riemann solver. The outputs are the resolved state and multidimensionally upwinded fluxes in both directions.

A formulation which respects the detailed geometry of the unstructured mesh is presented. Closed-form expressions are provided for all the integrals, making it particularly easy to implement the present multidimensional Riemann solvers in existing numerical codes. While it is visually demonstrated for three states coming together at a vertex, our formulation is general enough to treat multiple states (or zones with arbitrary geometry) coming together at a vertex. The present formulation is very useful for two-dimensional and three-dimensional unstructured mesh calculations of conservation laws. It has been demonstrated to work with second to fourth order finite volume schemes on two-dimensional unstructured meshes. On general triangular grids an arbitrary number of states might come together at a vertex of the primal mesh, while for calculations on the dual mesh usually three states come together at a grid vertex.

We apply the multidimensional Riemann solvers to hydrodynamics and magnetohydrodynamics (MHD) on unstructured meshes. The Riemann solver is shown to operate well for traditional second order accurate total variation diminishing (TVD) schemes as well as for weighted essentially non-oscillatory (WENO) schemes with ADER (Arbitrary DERivatives in space and time) time-stepping. Several stringent applications for compressible gasdynamics and magnetohydrodynamics are presented, showing that the method performs very well and reaches high order of accuracy in both space and time. The present multidimensional Riemann solver is cost-competitive with traditional, one-dimensional Riemann solvers. It offers the twin advantages of isotropic propagation of flow features and a larger CFL number.

Please see http://www.nd.edu/~dbalsara/Numerical-PDE-Course for a video introduction to multidimensional Riemann solvers.

I) Introduction

Riemann solvers are an essential ingredient of any Godunov scheme for hyperbolic conservation laws (Godunov [34], [35] van Leer [59]). Advances in computational fluid dynamics (CFD) have resulted in several useful Riemann solvers. Early work consisted of the exact Riemann solver by van Leer [59] and the approximate two-shock Riemann solver (Colella [22], Colella & Woodward [24]) and the random choice method by Chorin [20]. Approximate Riemann solvers have also been designed using rarefaction fans, see Osher and Solomon [45] and Dumbser & Toro [28]. Other very popular approximate Riemann solvers include the linearized Riemann solver by Roe [46] and the HLL/HLLE Riemann solvers (Harten, Lax & van Leer [37], Einfeldt [29], Einfeldt *et al*. [30]) and the local Lax-Friedrichs (LLF) Riemann solver (Rusanov [50]). Despite their robustness, HLL Riemann solvers suffer from the deficiency that they smear out contact discontinuities and shear waves, apart from the HLLEM scheme proposed in [30], where the constant intermediate state was substituted by a piecewise linear distribution that allowed to resolve the contact wave. An important breakthrough was made when Toro, Spruce and Speares [56] [57], [58], Chakraborty & Toro [19] and Batten *et al.* [15] showed how the contact discontinuity can be reintroduced in the HLL-type Riemann solvers, yielding a family of HLLC Riemann solvers. These one-dimensional HLLC Riemann solvers use a wave model that is much simpler than the full Riemann problem, yet they manage to capture the essential flow features and provide suitable entropy enforcement at rarefaction fans. Most importantly, they provide all these advantages at reasonably low computational cost. All the above-mentioned, one-dimensional Riemann solvers decompose discontinuities along the dominant directions of the mesh. Mesh imprinting, where the same flow features propagate at different speeds in different directions on a computational mesh, is an inevitable consequence of this approach. A CFL number that decreases as the dimensionality of the problem is increased is another unavoidable result.

Early attempts to introduce multidimensional effects into Riemann solvers were made in Roe [47] and Rumsey, van Leer & Roe [49], but they did not yield serviceable methods. As a result, an early trend consisted of using one-dimensional Riemann solvers in very intricate combinations in order to achieve multidimensional upwinding (Colella [23], Saltzman [51], LeVeque [39]). While these early efforts were not entirely true to the spirit of a genuinely multidimensional Riemann solver, the methods worked. The WAF scheme of Billett & Toro [16] represents another interesting line of development.

Abgrall [1], [2] formulated a genuinely multidimensional Riemann solver for CFD that worked. Pursuant to that work, further advances were also reported (Fey [31], [32], Gilquin, Laurens & Rosier [33], Brio, Zakharian & Webb [17], Lukacsova-Medvidova et al. [41]). All these Riemann solvers were multidimensional extensions of the linearized Riemann solver of Roe [46]. While intellectually appealing, they never saw wide-spread use in the CFD community; perhaps because they were specific to Euler flow and also perhaps because they were rather difficult to implement. Wendroff [60] formulated a two-dimensional HLL Riemann solver, but his method was also not easy to implement. Working on logically rectangular meshes, Balsara [4] presented a two-dimensional HLL Riemann solver with simple closed form expressions for the fluxes that were easy to implement. Balsara [5] subsequently presented a

two-dimensional HLLC Riemann solver which incorporated the physics of the contact discontinuity and permitted the contact discontinuity to propagate in any direction relative to the mesh. Balsara [5] also provided closed form expressions which made the two-dimensional HLLC Riemann solver easy to implement on any structured mesh, and for any hyperbolic conservation law. Examples drawn from Euler and MHD flow were presented to demonstrate that the method could take well to general systems of conservation laws. The paper by Balsara [5] also presents a video introduction to the method, which should make the subject more accessible to the general reader. While the work of Maire *et al.* [42] can be seen as a multidimensional Riemann solver for Lagrangian hydrodynamics, the present multidimensional Riemann solver can work on both, Arbitrary Lagrangian and Eulerian meshes and yields also a closed form formula for the mesh motion when used with a fully Lagrangian formulation. The goal of the present paper is to extend the formulation of Balsara [4] and Balsara [5] to unstructured meshes. As with previous papers in this series, we make a strong effort to present closed form expressions for the entire method, thus facilitating implementation. Several helpful appendices provide implementation-related details. The present formulation is applicable to any conservation law and is instantiated for the Euler and MHD equations.

In Balsara [5] we identified the essential principles for designing multidimensional, approximate Riemann solvers. For the sake of completeness, we adumbrate them here:

1) All one-dimensional Riemann solvers are self-similar when displayed on a space-time diagram. The same should be true when the wave model for a two-dimensional Riemann solver is displayed in two spatial and one temporal direction. In their simulations of multidimensional Riemann problems (using conventional, second-order Godunov schemes), Schulz-Rinne, Collins and Glaz [52] found that there exists a strongly interacting region where all the waves from all the different directions interact. They indeed verified that this region of strong interaction evolves in a self-similar fashion and we build that desirable property into the multidimensional Riemann solvers.

2) The Riemann solver should retain consistency with the conservation law. For the one-dimensional HLL Riemann solver, this means that the central state and the one-dimensional numerical flux are obtained by integrating the conservation law in one spatial and one temporal direction. For a multidimensional HLL Riemann solver, we obtain the strongly interacting state as well as the numerical fluxes in both directions by integrating the conservation law over well-chosen control volumes that are made of two spatial dimensions and one time dimension.

3) The central state in the one-dimensional HLL Riemann solver provides entropy enforcement in the vicinity of rarefaction fans. However, do please recall that the entropy enforcement only works if the wave speeds used in the wave model are the extremal ones. When designing a two-dimensional Riemann solver, we analogously take the strongly interacting state to be a constant state with the knowledge that it will provide the requisite amount of entropy enforcement. Just as before, this entropy enforcement requires that the bounding wavefronts in our wave model should be the extremal ones.

4) Recall that the one-dimensional HLLC Riemann solver was built on top of the one-dimensional HLL Riemann solver. The extremal waves remained unchanged. We simply introduced an additional wave for the contact discontinuity and required it to satisfy the jump conditions associated with a contact discontinuity. Designing a two-dimensional HLLC Riemann

solver requires us to follow the same strategy in multiple space dimensions. The extremally propagating wavefronts from the two-dimensional HLL Riemann solver are unchanged as one transitions to the two-dimensional HLLC Riemann solver. The two-dimensional HLL Riemann solver again yields the speed of the contact discontinuity. The orientation of the contact discontinuity can be gauged by identifying the direction of the density gradient on the computational mesh. Appropriate jump conditions are again asserted across the contact discontinuity and those conditions are consistent with the underlying conservation laws. Suitable space-time integration of the conservation law again yields the states and fluxes on either side of the contact discontinuity.

Magnetohydrodynamics (MHD) provides an interesting example of a conservation law with anisotropic wave motion. Consequently, it provides us with an interesting application domain for the present multidimensional Riemann solver technology. One-dimensional linearized Riemann solvers for numerical MHD have been designed (Roe & Balsara [48], Cargo and Gallice [18], Balsara [6]). HLLC Riemann solvers, capable of capturing mesh-aligned contact discontinuities, have been presented by Gurski [36] and Li [40]. Miyoshi and Kusano [43] drew on Gurski's work to design an HLLD Riemann solver for MHD. In this paper we demonstrate the performance of the two-dimensional HLLC Riemann solver for MHD problems on unstructured meshes.

We also verified via numerical experiments that our multidimensional HLLC Riemann solver works well with traditional TVD (total variation diminishing) and WENO (weighted esseintially non-oscillatory) schemes on unstructured meshes (Abgrall [3], Jiang & Shu [38], Balsara & Shu [8], Dumbser & Käser [26]), especially when the WENO reconstruction is coupled with a high order one-step ADER (Arbitrary DERivatives in space and time) time-evolution (Titarev & Toro [53], [54] and Toro & Titarev [55], Dumbser *et al*. [27], Balsara *et al*. [12], Balsara *et al*. [14]). ADER is a fully discrete approach in which both a high-order spatial representation of the data (reconstruction or any other, such as Discontinuous Galerkin schemes) and the time evolution are intimately linked via the generalised Riemann problem (generalized in the sense that the spatial representation of data is piece-wise smooth, eg. polynomials of any order, and the equations retain the source terms, if present). When used in most finite volume schemes, the multidimensional HLLC Riemann solver is made to operate on the primal triangular mesh. Using an extensive set of two-dimensional hydrodynamical test problems, we show that the ADER-WENO schemes operate robustly with CFL numbers that approach unity from below. Our ADER-WENO results even extend to MHD and we furthermore show in this paper that MHD simulations can also be run on unstructured meshes with CFL numbers approaching unity. Our multidimensional HLLC Riemann solver can be used on primal triangular meshes as well as on dual Voronoi tessellated meshes. This attests to the adaptability of our Riemann solver to all manner of unstructured meshes.

The plan of this paper is as follows. In Section II we formulate the multidimensional HLL Riemann solver on unstructured meshes and present a detailed examination of the multidimensional wave model in its subsonic and supersonic limits. Section III shows how the multidimensional HLLC Riemann solver is formulated on unstructured meshes. Closed form expressions for the strongly interacting states and numerical fluxes are presented in both of the previous sections. The correspondence between the two-dimensional HLL and HLLC Riemann solvers on structured and unstructured meshes is displayed in order to facilitate understanding.

Sections II and III provide details for the multidimensional HLL and HLLC Riemann solvers. Section IV synopsizes the steps required for implementing the multidimensional Riemann solvers. Section V presents accuracy analysis and presents a set of stringent test problems drawn from Euler flow that are successfully simulated on unstructured meshes using the present multidimensional Riemann solver technology. Section VI does the same for MHD flow. Section VII presents conclusions. Several helpful appendices are provided to facilitate implementation.

II) Multidimensional HLL Riemann Solver

Our formulation of the multidimensional HLL and HLLC Riemann solvers is general enough that it can, in principle, be applied to any multidimensional system of hyperbolic conservation laws. The present formulation represents an extension to unstructured meshes of the original insights developed in Balsara [5]. When applied to the Euler equations, we assume that the vector of conserved variables is ordered as density, *x,y,z*-momentum densities and energy density. The equation

$$\partial_t \mathbf{U} + \partial_x \mathbf{F} + \partial_y \mathbf{G} = 0 \tag{2.1}$$

describes a two-dimensional conservation law and serves to define the nomenclature for the conserved state $\mathbf{U}$ and the *x*- and *y*-fluxes, $\mathbf{F}$ and $\mathbf{G}$, respectively.

The multidimensional flow structure on a mesh is not visible at a zone face where only two states come together. Instead, it can be discerned at a vertex where multiple zones come together from all possible directions. Let "O" be the vertex between several zones that come together at that point. Let us locate the origin of our coordinate system at the point "O". Point "O" is also the vertex about which we want to solve the multidimensional HLL or HLLC Riemann problem. We focus on HLL first. Assume that three states $\mathbf{U}_1$, $\mathbf{U}_2$ and $\mathbf{U}_3$ come together at the vertex "O". The three states are arranged in an anticlockwise order as shown in Fig. 1. The unit vector $\boldsymbol{\eta}_1$ separates $\mathbf{U}_1$ and $\mathbf{U}_2$. The unit vector $\boldsymbol{\eta}_2$ separates $\mathbf{U}_2$ and $\mathbf{U}_3$; and the unit vector $\boldsymbol{\eta}_3$ separates $\mathbf{U}_3$ and $\mathbf{U}_1$. The unit vectors $\boldsymbol{\eta}_1$, $\boldsymbol{\eta}_2$ and $\boldsymbol{\eta}_3$ lie along the edges of our mesh and the three edges come together at the vertex "O". Let $\boldsymbol{\eta}_1 \equiv \eta_{1x}\mathbf{x} + \eta_{1y}\mathbf{y}$, with similar definitions for $\boldsymbol{\eta}_2 \equiv \eta_{2x}\mathbf{x} + \eta_{2y}\mathbf{y}$ and $\boldsymbol{\eta}_3 \equiv \eta_{3x}\mathbf{x} + \eta_{3y}\mathbf{y}$. (Here $\mathbf{x}$ and $\mathbf{y}$ are unit vectors in the *x*- and *y*-directions. The unit vector $\mathbf{t}$ is parallel to the time axis in a three-dimensional space-time.) We refer to these three unit vectors as the principal directions of the mesh at the point "O". At each of the edges that come together at "O", we wish to build a local basis in two-dimensions. Consequently, associated with $\boldsymbol{\eta}_1$ we have $\boldsymbol{\tau}_1$ such that $\boldsymbol{\eta}_1 \bullet \boldsymbol{\tau}_1 = 0$, with the result that $\boldsymbol{\tau}_1 = -\eta_{1y}\mathbf{x} + \eta_{1x}\mathbf{y}$. Similar definitions hold for $\boldsymbol{\tau}_2$ and $\boldsymbol{\tau}_3$ so that $\boldsymbol{\eta}_2 \bullet \boldsymbol{\tau}_2 = 0$ and $\boldsymbol{\eta}_3 \bullet \boldsymbol{\tau}_3 = 0$. We therefore have $\boldsymbol{\tau}_2 = -\eta_{2y}\mathbf{x} + \eta_{2x}\mathbf{y}$ and $\boldsymbol{\tau}_3 = -\eta_{3y}\mathbf{x} + \eta_{3x}\mathbf{y}$. Notice that the states and unit vectors are chosen so that they have a counterclockwise orientation. As long as that sense of direction is preserved, the subsequent formulae are designed to work automatically.

The waves emanating from "O" over a time interval "*T*" are all contained within $\Delta P_1P_2P_3$, as shown in Fig. 1. As a result, the wavefront P_1P_2 propagates with a speed S_1 in the direction $\boldsymbol{\eta}_1$. Similarly, the other two wavefronts, P_2P_3 and P_3P_1, propagate with a speeds S_2 and S_3 in the directions $\boldsymbol{\eta}_2$ and $\boldsymbol{\eta}_3$. Together, these wavefronts serve to circumscribe the strongly interacting state.

Please note that while Fig. 1 displays just three states coming together at a vertex, the method is general and can accommodate any number of states coming together at the vertex "O". Thus let "*imax*" be the maximum number of states, where the set of states is given by $\{\mathbf{U}_i \; ; \; i = 1,..,imax\}$. We require that the states $\mathbf{U}_i$ are arranged in a counterclockwise orientation. Let $\boldsymbol{\eta}_i \equiv \eta_{ix}\mathbf{x} + \eta_{iy}\mathbf{y}$ be the unit vector separating the states $\mathbf{U}_i$ and $\mathbf{U}_{cyc(i+1)}$. (Here the cyclic function is defined as $cyc(i) = i$ when $1 \le i \le imax$ and $cyc(i) = i - imax$ when $imax < i \le 2\ imax$.) For each unit normal, $\boldsymbol{\eta}_i$, we define a unit vector $\boldsymbol{\tau}_i \equiv -\eta_{iy}\mathbf{x} + \eta_{ix}\mathbf{y}$ that is orthogonal to it. We, therefore, see that these unit vectors, $\boldsymbol{\eta}_i$ and $\boldsymbol{\tau}_i$, carry information about the mesh geometry. Since the present multidimensional Riemann solvers incorporate this information about the mesh geometry into the solution procedure, we see that multidimensional effects associated with mesh geometry are intimately woven into the solution of our multidimensional Riemann solvers. As in Fig. 1, we have a wavefront $P_iP_{cyc(i+1)}$ propagating with speed S_i along each principal direction $\boldsymbol{\eta}_i$. To keep the wave model simple, we make $P_iP_{cyc(i+1)}$ orthogonal to $\boldsymbol{\eta}_i$. As before, these wavefronts circumscribe the strongly interacting state. Also notice that as the number of triangles coming together at a vertex increases, i.e. as the angular resolution of the underlying mesh increases, this procedure yields a wave model that increasingly approximates a Monge cone.

The further description of the multidimensional HLL Riemann solver is most easily understood if it is first sketched out pictorially in this paragraph. Based on the pictorial sketch, which we present in pointwise form, we identify three tasks that we need to carry out in order to formulate a multidimensional HLL Riemann solver. The three sub-sections that follow this pictorial study provide further details about those three tasks. For some of those tasks we have supplied helpful appendices that provide the implementation-related details. The three tasks are:

1) Solve One-Dimensional Riemann Problems Perpendicular to the Principal Directions: Fig. 2a shows the space-time extension of Fig. 1. We see that $\Delta P_1P_2P_3$ from Fig. 1 has now become a triangular prism in space-time with its footpoints at $\Delta P_1'P_2'P_3'$. The darkly shaded triangular areas on the side panels of Fig. 2a are indeed the resolved states between the one-dimensional Riemann problems. Thus we will have to solve one-dimensional Riemann problems between states $\mathbf{U}_i$ and $\mathbf{U}_{cyc(i+1)}$. Notice that these Riemann problems are being solved in the direction $\boldsymbol{\tau}_i$ which is perpendicular to the principal direction $\boldsymbol{\eta}_i$. Various fluxes, both in the $\boldsymbol{\tau}_i$ and $\boldsymbol{\eta}_i$ directions, will be sought from the solution of these one-dimensional Riemann problems. Notice too that these Riemann problems are to be solved in rotated frames of reference. As a result, Sub-section II.1 will establish the nomenclature and provide further detail. Appendix A

shows how this task is implemented with a minimal number of coordinate transformations for HLL and HLLC Riemann solvers for Euler flow.

2) Use the Wave Speeds from the One-Dimensional Riemann Problems to Obtain the Multidimensional Wave Model: It is well-known that the entropy enforcement at rarefaction fans will not be adequate in a one-dimensional HLL Riemann solver unless the maximal left- and right-going wave speeds are selected to be the lower and upper bounds of all the waves emanating from the original discontinuity. Also recall that the subsonic state is bounded by these extremal speeds. In an entirely analogous fashion, the extremal speeds in the principal directions are shown in Fig. 1. The strongly interacting state $\mathbf{U}^*$, shown in Fig. 2a, should also be such that it fully contains all the wave speeds emanating from the origin "O" from all the different one-dimensional Riemann problems. In other words, $\Delta P_1P_2P_3$ is the envelope of all the waves that could conceivably emanate in all possible directions from the origin. The extent of the strongly interacting state in space-time is shown in Fig. 2b. If it were possible to remove the side panels in Fig. 2a to expose the strongly interacting state, we would indeed see Fig. 2b. Notice that the inverted triangular pyramid in Fig. 2b forms a self-similar structure in space-time. As shown in Fig. 1, the extremal wavefronts move with speed S_i along each principal direction $\boldsymbol{\eta}_i$. Specifying S_i would indeed specify the multidimensional wave model. Further details are provided in Sub-section II.2 and a computationally implementable, three-step strategy for obtaining the speeds S_i is provided in Appendix B.

3) Obtain the Strongly Interacting State and Associated Numerical Fluxes: If $\Delta P_1P_2P_3$ overlies the time axis in Fig. 2b, we say that the strongly interacting state, $\mathbf{U}^*$, is multidimensionally subsonic. As with the one-dimensional HLL Riemann solver, we can obtain $\mathbf{U}^*$ by integrating the two-dimensional conservation law over the three-dimensional inverted triangular pyramid shown in Fig. 2b. We will obtain the associated numerical fluxes in the *x*- and *y*-directions via further integrations of the two-dimensional conservation law over sub-portions of the volume shown in Fig. 2b. This is detailed in Sub-section II.3. If $\Delta P_1P_2P_3$ does not overlie the time axis, a situation that is easily identified because one of the speeds S_i will then be negative in Fig. 1, then we say that the problem is supersonic. Sub-section II.3 provides further details associated with obtaining the multidimensionally upwinded, supersonic numerical fluxes.

II.1) Solving One-Dimensional Riemann Problems Perpendicular to the Principal Directions

We see from Fig. 2a that between the states $\mathbf{U}_1$ and $\mathbf{U}_2$, we can solve a one-dimensional HLL Riemann problem. (I.e., in the traditional sense, $\mathbf{U}_1$ and $\mathbf{U}_2$ are the left and right states of the one-dimensional Riemann problem. However, note that when that convention is used, $\boldsymbol{\tau}_1$ plays the role of the positive *x*-axis while $\boldsymbol{\eta}_1$ is in the direction of the negative *y*-axis.) In actuality, it helps to define a rotated coordinate frame (η_1, τ_1) and to directly solve the one-

dimensional Riemann problem in the $\boldsymbol{\tau}_1$ direction. In that rotated frame, the conservation law from eqn. (2.1) becomes

$$\partial_t \mathbf{U} + \partial_{\eta 1} \mathbf{G}_{\boldsymbol{\eta} 1} + \partial_{\tau 1} \mathbf{F}_{\boldsymbol{\tau} 1} = 0 \tag{2.2}$$

(The subscripts of the fluxes $\mathbf{G}_{\boldsymbol{\eta} 1}$ and $\mathbf{F}_{\boldsymbol{\tau} 1}$ in the above equation indicate the directions in which they act.) Thus the extremal waves in this Riemann problem propagate orthogonal to $\boldsymbol{\eta}_1$; i.e., in the direction $\boldsymbol{\tau}_1$. Call these extremal speeds $S_{\boldsymbol{\tau} 1}^{+}$ and $S_{\boldsymbol{\tau} 1}^{-}$ along $\boldsymbol{\tau}_1$. This HLL Riemann problem produces the intermediate HLL state $\mathbf{U}_1^*$ and associated flux $\mathbf{F}_{\boldsymbol{\tau} 1}^*$ in the direction $\boldsymbol{\tau}_1$. Fig. 2a shows this situation. Using this intermediate state and flux, we can obtain the transverse flux $\mathbf{G}_{\boldsymbol{\eta} 1}^*$ which represents the flux of fluid variables in the direction $\boldsymbol{\eta}_1$. Appendix A provides details for obtaining the intermediate states and fluxes for HLL and HLLC Riemann solvers with a minimal amount of coordinate transformations.

The generalization from Fig. 2a is that we focus on two states $\mathbf{U}_i$ and $\mathbf{U}_{cyc(i+1)}$. We work in the rotated coordinate frame (η_i, τ_i) which is defined by the unit vectors $\boldsymbol{\eta}_i$ and $\boldsymbol{\tau}_i$. In that frame, the extremal speeds $S_{\boldsymbol{\tau} i}^{+}$ and $S_{\boldsymbol{\tau} i}^{-}$ along $\boldsymbol{\tau}_i$ serve to bound the intermediate HLL state $\mathbf{U}_i^*$. Corresponding to $\mathbf{U}_i^*$, we can find the fluxes $\mathbf{F}_{\boldsymbol{\tau} i}^*$ and $\mathbf{G}_{\boldsymbol{\eta} i}^*$.

Because this paper focuses on HLLC Riemann solvers, we will in fact want the solution of the one-dimensional HLLC Riemann problem in the rotated coordinates (η_i, τ_i). Appendix A shows us how the solutions of the one-dimensional HLL and HLLC Riemann problems are most easily obtained in those rotated coordinates. This completes our description of the one-dimensional Riemann problems that are solved perpendicular to the principal directions.

II.2) Using the Wave Speeds from the One-Dimensional Riemann Problems to Obtain the Multidimensional Wave Model

The one dimensional Riemann problem between the states $\mathbf{U}_1$ and $\mathbf{U}_2$ enables us to extract the extremal speeds $S_{\boldsymbol{\tau} 1}^{+}$ and $S_{\boldsymbol{\tau} 1}^{-}$ in the $\boldsymbol{\tau}_1$ direction, see Fig. 2a. The availability of the intermediate state $\mathbf{U}_1^*$ and its associated flux can be used to obtain a corresponding velocity, pressure and density in that intermediate state (see Appendix A). This information can be used to obtain the speeds $S_{\boldsymbol{\eta} 1}^{+}$ and $S_{\boldsymbol{\eta} 1}^{-}$ in the $\boldsymbol{\eta}_1$ direction. Fig. 3a shows us how the speeds $S_{\boldsymbol{\tau} 1}^{+}$, $S_{\boldsymbol{\tau} 1}^{-}$, $S_{\boldsymbol{\eta} 1}^{+}$ and $S_{\boldsymbol{\eta} 1}^{-}$ can be used to establish an ellipse E_1 in the rotated frame defined by the unit vectors $\boldsymbol{\eta}_1$ and $\boldsymbol{\tau}_1$. The boundary of the ellipse E_1 traces out speeds of wave propagation in that rotated frame. This ellipse may be offset from the origin. For subsonic flow, the ellipse intersects the axes shown in Fig. 3a at four points and those points provide sufficient information for fully

specifying the ellipse. For hydrodynamics, this ellipse becomes a circle with a center that is offset from the origin by the fluid velocity. If we examine how this circle evolves in time, we realize that it traces out a Monge cone in space and time. (For anisotropic problems, like MHD, the extremal speeds form an ellipse that is bounded by the fast magnetosonic speed in each direction and the ellipse is offset from the origin by the flow velocity.) Such an ellipse is also shown in Fig. 3b and it will be used to identify the surface of maximal wave propagation for the multidimensional Riemann problem.

When three zones come together at a vertex, $\Delta P_1P_2P_3$ in Fig. 1 and Fig. 2 identifies the surface of maximal wave propagation; i.e., the multidimensional wave model. In other words, the wave model is the triangular envelope of the three ellipses E_1 , E_2 and E_3. This triangle is formed by wavefronts P_1P_2 , P_2P_3 and P_3P_1 which propagate with speeds S_1 , S_2 and S_3 along the principal directions of the mesh. Specifying the speeds S_1 , S_2 and S_3 along the principal directions of the mesh, therefore, specifies the multidimensional wave model.

In the more general case, the one dimensional Riemann problem between the states $\mathbf{U}_i$ and $\mathbf{U}_{cyc(i+1)}$ enables us to obtain the ellipse E_i shown in Fig. 3a. The ellipse E_i is obtained by identifying the extremal speeds $S^+_{\boldsymbol{\tau}i}$ and $S^-_{\boldsymbol{\tau}i}$ in the $\boldsymbol{\tau}_i$ direction and the speeds $S^+_{\boldsymbol{\eta}i}$ and $S^-_{\boldsymbol{\eta}i}$ in the $\boldsymbol{\eta}_i$ direction. The envelope of this full set of ellipses $\{E_i; i = 1,..,imax\}$ will form an *imax*-sided polygon. The polygon is specified by specifying the speeds S_i along the principal directions $\boldsymbol{\eta}_i$. Our goal in this Sub-section is to find the speeds S_i.

It is important to realize that it is difficult to parametrize the boundary of an ellipse and use it to find the envelope of several ellipses. As a result, we simplify the problem, as shown in Fig. 3b. Given the speeds $S^+_{\boldsymbol{\tau}i}$, $S^-_{\boldsymbol{\tau}i}$, $S^+_{\boldsymbol{\eta}i}$ and $S^-_{\boldsymbol{\eta}i}$ for the ellipse E_i, we can find the smallest rectangle R_i which contains the ellipse. Just like the boundary of E_i , the boundary of the rectangle R_i traces out a speed of propagation in each rotated frame shown in Fig. 3b. Finding an *imax*-sided polygon that contains the set of rectangles $\{R_i; i = 1,..,imax\}$ is easy, because we only have to ensure that the four vertices of each rectangle R_i lie within the desired polygon.

The detailed task of finding the speeds S_i along the principal directions $\boldsymbol{\eta}_i$ can be implemented in two simple steps on a computer. In the first step, we establish the wave speeds S_i along each principal direction $\boldsymbol{\eta}_i$ by projecting the four vertices of all the rectangles, $R_j : j = 1,..,imax$, onto that principal direction and taking the maximum of those projected speeds. We also need to ensure that each face of the resulting polygon contains the entire one-dimensional wave model in its local $\boldsymbol{\tau}$-direction. This is done in the second step where we further expand the wave speeds in the principal directions so that the one-dimensional wave model for each one-dimensional Riemann problem is fully contained within the corresponding side panel of Fig. 2a. The computer-friendly implementation of these two steps has been catalogued in detail in Appendix B. The construction presented in Appendix B is such that it retrieves the wave

model of Balsara [5] for Cartesian meshes. We are, therefore, reassured that there is at least one multidimensional limit in which it matches with prior work.

Once the speeds S_i along the principal directions $\boldsymbol{\eta}_i$ are found, we can easily determine whether the multidimensional Riemann problem is subsonic or supersonic. If all the speeds S_i are positive, the multidimensional Riemann problem is subsonic because it covers the time axis; otherwise it is supersonic.

Recall that the wave speeds for the one-dimensional HLL Riemann solver can be written in such a way that the subsonic case retrieves all the desired fluxes. In an entirely analogous fashion, Appendix B also shows us how to do the same for the multi-dimensional Riemann solver. Thus Appendix B presents two variants of the wave model. In the first variant of the wave model, the strongly interacting state can overlie the time axis to yield the subsonic case or it may not overlie the time axis to yield the supersonic case. In the second variant of the wave model, we modify the wave model so that the same subsonic case retrieves all the desired fluxes. As expected, the second variant of the wave model is obtained by expanding the first variant.

The extent of the strongly interacting state in space-time is shown in Fig. 2b, which also displays the self-similar nature of our wave model. The three grey lines in Fig. 1 show the bounding wavefronts of the multidimensional Riemann problem. They have loci

$$y = S_1 T\ \eta_{1y} - \frac{\eta_{1x}}{\eta_{1y}}\left(x - S_1 T\ \eta_{1x}\right)\ ;\ y = S_2 T\ \eta_{2y} - \frac{\eta_{2x}}{\eta_{2y}}\left(x - S_2 T\ \eta_{2x}\right)\ ;\ y = S_3 T\ \eta_{3y} - \frac{\eta_{3x}}{\eta_{3y}}\left(x - S_3 T\ \eta_{3x}\right)$$

(2.3)

The intersection points of these three lines (wavefronts) are useful to our derivation. Let those points, along with their coordinates in a three-dimensional space-time (two space and one time dimension), be denoted by $\mathrm{P}_1 \equiv \left(x_1 T, y_1 T, T\right)$, $\mathrm{P}_2 \equiv \left(x_2 T, y_2 T, T\right)$ and $\mathrm{P}_3 \equiv \left(x_3 T, y_3 T, T\right)$, as shown in Fig. 1 and Fig. 2. Notice that $\left(x_i, y_i\right)$ have units of speed. To help with computer implementation, we explicitly write:

$$x_1 = \frac{S_3\eta_{1y} - S_1\eta_{3y}}{\eta_{3x}\eta_{1y} - \eta_{3y}\eta_{1x}}\ ;\ y_1 = -\frac{S_3\eta_{1x} - S_1\eta_{3x}}{\eta_{3x}\eta_{1y} - \eta_{3y}\eta_{1x}}\ ;\ x_2 = \frac{S_1\eta_{2y} - S_2\eta_{1y}}{\eta_{1x}\eta_{2y} - \eta_{1y}\eta_{2x}}\ ;\ y_2 = -\frac{S_1\eta_{2x} - S_2\eta_{1x}}{\eta_{1x}\eta_{2y} - \eta_{1y}\eta_{2x}}\ ;$$
$$x_3 = \frac{S_2\eta_{3y} - S_3\eta_{2y}}{\eta_{2x}\eta_{3y} - \eta_{2y}\eta_{3x}}\ ;\ y_3 = -\frac{S_2\eta_{3x} - S_3\eta_{2x}}{\eta_{2x}\eta_{3y} - \eta_{2y}\eta_{3x}}\ ;$$

(2.4)

Using the above formulae, the lengths $l_1^- T$ and $l_1^+ T$ in Fig. 1 are easily evaluated. Similarly, we can evaluate $l_2^- T$, $l_2^+ T$, $l_3^- T$ and $l_3^+ T$ in Fig. 1. By design, l_1^- , l_1^+ , l_2^- , l_2^+ , l_3^- and l_3^+ have units of speed. These six speeds give us the limiting speeds from the one-dimensional Riemann problems that can be represented within the grey triangle in Fig. 1. Fig. 4 schematically shows us those limiting speeds as well as the speeds associated with each of the one-dimensional Riemann

problems in each of the faces. Fig. 4 will become very useful when evaluating space-time integrals in the next Sub-section. This Sub-section has, therefore, detailed all the geometric constructions that are needed for evaluating the strongly interacting state $\mathbf{U}^*$ in Fig. 2b. Lastly, notice that the construction is entirely general and can be extended naturally to the case where more than three states converge at the vertex "O" in Fig. 1.

II.3) Obtaining the Strongly Interacting State and Associated Numerical Fluxes

This Sub-section is divided into four parts. The first part details the derivation of the multidimensional HLL state $\mathbf{U}^*$ in Sub-section II.3.a. The second part details the derivation of the multidimensional HLL *x*-flux $\mathbf{F}^*$ in the subsonic limit in Sub-section II.3.b. The third part details the derivation of the multidimensional HLL *y*-flux $\mathbf{G}^*$ in the subsonic limit in Sub-section II.3.c. Sub-section II.3.d provides the formulae for the numerical fluxes in the supersonic case. We also show the correspondence between the two-dimensional HLL Riemann solvers on structured and unstructured meshes in order to facilitate understanding. The formulae in this Sub-section follow the same sequence that one should use in making a computer implementation of the two-dimensional HLL Riemann solver. This should simplify the process of making a computer implementation.

II.3.a) Obtaining the Strongly Interacting HLL State:

The strongly interacting HLL state $\mathbf{U}^*$ is obtained by integrating the conservation law in eqn. (2.1) over the inverted pyramid shown in Fig. 2b. All the geometric constructions required for finding the boundaries of that pyramid have been worked out in Sub-section II.2 where the wave model was catalogued. The total area of the triangle $\Delta \mathrm{P_1P_2P_3}$ in Fig. 1 is also useful, and it is also easy to find. In general, the top of the pyramid in Fig. 2b will be a polygon. Its area, after an obvious T^2 scaling, is given by a series of cross products in the two-dimensional *xy*-plane, as:

$$\mathrm{A}_{UHLL} T^2 = \frac{1}{2} \sum_{i=1}^{imax} \overrightarrow{\mathrm{OP}_i} \times \overrightarrow{\mathrm{OP}_{cyc(i+1)}}$$

$$\mathrm{A}_{UHLL} = \frac{1}{2} \sum_{i=1}^{imax} \left(x_i y_{cyc(i+1)} - y_i x_{cyc(i+1)} \right) \qquad (2.5)$$

As long as the "*imax*" number of zones are arranged in an anti-clockwise, cyclic order, the above formula should work well in the subsonic and supersonic limits.

To obtain $\mathbf{U}^*$ in all circumstances, subsonic and supersonic, we carry out a space-time integration of the conservation law over the volume of the inverted pyramid shown in Fig. 2b. Gauss Law is then used to turn the volumetric integral into a set of area integrals applied to the

boundaries of the inverted pyramid. In eqn. (2.5) we have already evaluated the area integral over the base of the inverted pyramid. The area integral over any of the side panels of Fig. 2b is more difficult, and we further explain the process of obtaining one of those integrals. The speeds associated with the one-dimensional HLL Riemann solvers can be mapped into those local coordinate systems, as shown in Fig. 4. Fig. 4 essentially shows all the speeds arising from all the one-dimensional HLL Riemann solvers in all the faces of Fig. 2a. Fig. 5 further reinforces that concept by showing us the 1st side panel from Fig. 2b. If the shaded $\Delta \mathrm{OP}_a\mathrm{P}_b$ is a constant state in that panel then its area vector in space-time is given by the vector cross product $\overrightarrow{\mathrm{OP}_a} \times \overrightarrow{\mathrm{OP}_b}/2$. The unit vectors $\boldsymbol{\eta}_1$ and $\boldsymbol{\tau}_1$ along with the speeds S_1 and l_a in Fig. 5 can be used to find the space-time location of the point $\mathrm{P_a}$; we do similarly for the point $\mathrm{P_b}$. When we take the inner product of the previous area vector with $\mathbf{F\,x + G\,y + U\,t}$, we obtain the requisite area integral of the conservation law. Please see Appendix C for more detail.

Insights from the previous two paragraphs and Appendix C now show their usefulness in obtaining the strongly interacting state $\mathbf{U}^*$. The integral over the side panels can be generalized and written in closed form, with an obvious T^2 scaling, as

$$\frac{1}{T^2}\Big(\text{Area integral over } \mathrm{OP}_i\mathrm{P}_{cyc(i+1)}\Big) =$$
$$\frac{1}{2}\Big[\Big(S_{\boldsymbol{\tau}i}^- - l_i^-\Big)\Big(\mathbf{G}_{\boldsymbol{\eta}i} - S_i\mathbf{U}_i\Big) + \Big(S_{\boldsymbol{\tau}i}^+ - S_{\boldsymbol{\tau}i}^-\Big)\Big(\mathbf{G}_{\boldsymbol{\eta}i}^* - S_i\mathbf{U}_i^*\Big) + \Big(l_i^+ - S_{\boldsymbol{\tau}i}^+\Big)\Big(\mathbf{G}_{\boldsymbol{\eta}cyc(i+1)} - S_i\mathbf{U}_{cyc(i+1)}\Big)\Big] \quad (2.6)$$

The fluxes $\mathbf{G}_{\boldsymbol{\eta}i}$ and $\mathbf{G}_{\boldsymbol{\eta}cyc(i+1)}$ in the above equation correspond to the states $\mathbf{U}_i$ and $\mathbf{U}_{cyc(i+1)}$ respectively, and they are oriented in the $\boldsymbol{\eta}_i$ direction. The flux $\mathbf{G}_{\boldsymbol{\eta}i}^*$ in the above equation corresponds to the intermediate state $\mathbf{U}_i^*$ from the one-dimensional HLL Riemann solver and is also oriented in the $\boldsymbol{\eta}_i$ direction. After space-time integration of the conservation law over the inverted pyramid shown in Fig. 2b we get

$$\mathrm{A}_{UHLL}\mathbf{U}^* + \sum_{i=1}^{imax}\frac{1}{T^2}\Big(\text{Area integral over } \mathrm{OP}_i\mathrm{P}_{cyc(i+1)}\Big) = 0 \quad (2.7)$$

which gives us

$$\mathbf{U}^* = -\frac{1}{\mathrm{A}_{UHLL}}\left\{\sum_{i=1}^{imax}\frac{1}{T^2}\Big(\text{Area integral over } \mathrm{OP}_i\mathrm{P}_{cyc(i+1)}\Big)\right\} \quad (2.8)$$

This gives us a closed form expression for the strongly interacting state $\mathbf{U}^*$ in the multidimensional HLL Riemann solver. Eqn. (2.8) is one of the most important equations in this paper because it expresses the strongly interacting subsonic state $\mathbf{U}^*$ in terms of panel integrals taken over the sides of our wave model. Please notice that the final formula in eqn. (2.8) is so designed that the time T drops out of the equation; see eqn. (2.6) to see why. This will also be true for the final expressions of all the numerical fluxes. The self-similarity of the Riemann problem ensures that the final expressions should be free of the time variable.

II.3.b) Obtaining the Multidimensional HLL x-Flux in the Subsonic limit:

The present multidimensional construction for the numerical fluxes is only needed for the subsonic case. In order to obtain the *x*-flux in the specific case that is shown in Fig. 2b, we need to integrate over a sub-portion of the strongly interacting state $\mathbf{U}^*$ that includes the plane $x=0$. The $\Delta \mathrm{OQ_1Q_2}$ in Fig. 2b explicitly shows this plane. The volume in space-time over which we integrate the conservation law in eqn. (2.1) is then given by the sub-portion of the strongly interacting state that has $x \le 0$. For the specific case shown in Fig. 2b, this volume is given by the inverted pyramid with a base given by $\Delta \mathrm{Q_1Q_2P_3}$ and an apex given by "O". Fig. 6a shows the more general case where only the base of the inverted pyramid is shown for the situation where the strongly interacting state is bounded by a pentagon; i.e. five zones come together at that vertex. The thick lines of Fig. 6a identify the sub-portion of the pentagon that will be integrated over. In automating the process of integrating the conservation laws over these surfaces, we have to automate the process of identifying the intersection of the plane $x=0$ with the wavefronts that make up the boundary of Fig. 6a.

Wavefronts that lie entirely to the left of $x=0$ are easily identified from Fig. 6a because both their end-points in eqn. (2.4) will have *x*-coordinates that are less than or equal to zero. I.e., we say that the i^{th} side panel will contribute fully to the area integral by evaluating the condition $\left(x_i \le 0\right) and \left(x_{cyc(i+1)} \le 0\right)$. The panel integrals over the side panels of the inverted pyramid are easily obtained for those wavefronts by using eqn. (2.6). Wavefronts which have both end points with *x*-coordinates that are greater than or equal to zero will not contribute to the integrals over the side panels of the inverted pyramid.

For each of the remaining wavefronts in eqn. (2.3), we find the intersection of the plane $x=0$ with those wavefronts. The point of intersection of $x=0$ with the wavefront $\mathrm{P}_i\mathrm{P}_{cyc(i+1)}$ is given by $\mathrm{P}_{i;X} = \left(0, Y_i T, T\right)$ where $Y_i = S_i / \eta_{iy}$. Owing to the convexity of the wave model in the subsonic case, there are exactly two such wavefronts which have an interior intersection with the plane $x=0$. Let us identify them with indicial labels "*i*=*istart_x*" and "*i*=*iend_x*". In practice, "*istart_x*" is identified by the condition $\left(x_{istart\_x} > 0\right) and \left(x_{cyc(istart\_x+1)} \le 0\right)$. Likewise, in practice, "*iend_x*" is identified by the condition $\left(x_{iend\_x} \le 0\right) and \left(x_{cyc(iend\_x+1)} > 0\right)$. Please see Fig. 6a for a good demonstration of how one obtains the indices "*i*=*istart_x*" and "*i*=*iend_x*".

Fig. 6a, is to be viewed as having a predominantly anti-clockwise orientation. Notice from Fig. 6a that for the side panel with "*i*=*istart_x*" we will be integrating from some intermediate point all the way to the end of that panel. We have already seen how the speeds l_i^- and l_i^+ in Fig. 1 and Fig. 2a form a local coordinate system in the i^{th} side panel. The speeds associated with the one-dimensional HLL Riemann solvers can be mapped into those local coordinate systems, as shown in Fig. 4. Fig. 5 further reinforces that concept by showing us the

1st side panel from Fig. 2b. It shows how the integrals can be carried out over the side panels of the inverted pyramids by focusing on triangles in those side panels that correspond to constant states. As a result, by using $Y_{istart\_x} = S_{istart\_x} / \eta_{istart\_x;y}$ we can obtain a speed l_{istart_x} which identifies the lower limit of integration in the side panel with "*i=istart_x*". After an obvious T^2 scaling, we use the results from Appendix C to integrate over the "*istart_x*" side panel from l_{istart_x} to $l^+_{istart\_x}$. We get:

$$\frac{1}{T^2}\left(\text{Area integral over OP}_{cyc(istart\_x+1)}\text{P}_{istart\_x;X}\right) =$$
$$\frac{1}{2}\left[\begin{array}{l} \max\left(S^-_{\boldsymbol{\tau} istart\_x} - l_{istart\_x}, 0\right)\left(\mathbf{G}_{\boldsymbol{\eta} istart\_x} - S_{istart\_x}\mathbf{U}_{istart\_x}\right) \\ +\left(S^+_{\boldsymbol{\tau} istart\_x} - \max\left(S^-_{\boldsymbol{\tau} istart\_x}, l_{istart\_x}\right)\right)\text{H}\left(S^+_{\boldsymbol{\tau} istart\_x} - l_{istart\_x}\right)\left(\mathbf{G}^*_{\boldsymbol{\eta} istart\_x} - S_{istart\_x}\mathbf{U}^*_{istart\_x}\right) \\ +\left(l^+_{istart\_x} - \max\left(S^+_{\boldsymbol{\tau} istart\_x}, l_{istart\_x}\right)\right)\left(\mathbf{G}_{\boldsymbol{\eta} cyc(istart\_x+1)} - S_{istart\_x}\mathbf{U}_{cyc(istart\_x+1)}\right) \end{array}\right] \quad (2.9)$$

The fluxes $\mathbf{G}_{\boldsymbol{\eta} istart\_x}$ and $\mathbf{G}_{\boldsymbol{\eta} cyc(istart\_x+1)}$ in the above equation correspond to the states $\mathbf{U}_{istart\_x}$ and $\mathbf{U}_{cyc(istart\_x+1)}$ respectively, and they are oriented in the $\boldsymbol{\eta}_{istart\_x}$ direction. The flux $\mathbf{G}^*_{\boldsymbol{\eta} istart\_x}$ in the above equation corresponds to the intermediate state $\mathbf{U}^*_{istart\_x}$ from the one-dimensional HLL Riemann solver and is also oriented in the $\boldsymbol{\eta}_{istart\_x}$ direction. (Here H(x) is the Heaviside function; 1 for $x \geq 0$ and 0 for x<0.)

Using Fig. 6a and $Y_{iend\_x} = S_{iend\_x} / \eta_{iend\_x;y}$, we can obtain a speed l_{iend_x} which identifies the upper limit of integration in the side panel with "*i=iend_x*". We also use the results from Appendix C to integrate over the "*iend_x*" side panel from $l^-_{iend\_x}$ to l_{iend_x}. We get:

$$\frac{1}{T^2}\left(\text{Area integral over OP}_{iend\_x;X}\text{P}_{iend\_x}\right) =$$
$$\frac{1}{2}\left[\begin{array}{l} \left(\min\left(S^-_{\boldsymbol{\tau} iend\_x}, l_{iend\_x}\right) - l^-_{iend\_x}\right)\left(\mathbf{G}_{\boldsymbol{\eta} iend\_x} - S_{iend\_x}\mathbf{U}_{iend\_x}\right) \\ +\left(\min\left(S^+_{\boldsymbol{\tau} iend\_x}, l_{iend\_x}\right) - S^-_{\boldsymbol{\tau} iend\_x}\right)\text{H}\left(l_{iend\_x} - S^-_{\boldsymbol{\tau} iend\_x}\right)\left(\mathbf{G}^*_{\boldsymbol{\eta} iend\_x} - S_{iend\_x}\mathbf{U}^*_{iend\_x}\right) \\ +\max\left(l_{iend\_x} - S^+_{\boldsymbol{\tau} iend\_x}, 0\right)\left(\mathbf{G}_{\boldsymbol{\eta} cyc(iend\_x+1)} - S_{iend\_x}\mathbf{U}_{cyc(iend\_x+1)}\right) \end{array}\right] \quad (2.10)$$

The fluxes $\mathbf{G}_{\boldsymbol{\eta} iend\_x}$ and $\mathbf{G}_{\boldsymbol{\eta} cyc(iend\_x+1)}$ in the above equation correspond to the states $\mathbf{U}_{iend\_x}$ and $\mathbf{U}_{cyc(iend\_x+1)}$ respectively, and they are oriented in the $\boldsymbol{\eta}_{iend\_x}$ direction. The flux $\mathbf{G}^*_{\boldsymbol{\eta} iend\_x}$ in the above equation corresponds to the intermediate state $\mathbf{U}^*_{iend\_x}$ from the one-dimensional HLL Riemann solver and is also oriented in the $\boldsymbol{\eta}_{iend\_x}$ direction. Along with eqn. (2.6), eqns. (2.9) and (2.10) give us all the area integrals that we need over the side panels of Fig. 2b for the construction of the HLL x-flux.

Notice that in the specific case of Fig. 2b, we will also need the area of $\Delta Q_1 Q_2 P_3$. In general, as seen from Fig. 6a, the area bounded by the thick line in that figure is a polygon. We call that area A_{FHLL}. After a T^2 scaling, it is given by a series of cross products in the two-dimensional *xy*-plane, as:

$$A_{FHLL}T^2 = \frac{1}{2}\left|\overrightarrow{OP_{istart\_x;X}} \times \overrightarrow{OP_{cyc(istart\_x+1)}}\right| + \frac{1}{2}\left|\overrightarrow{OP_{iend\_x}} \times \overrightarrow{OP_{iend\_x;X}}\right| + \frac{1}{2}\sum_{i=1}^{imax}\left|\overrightarrow{OP_i} \times \overrightarrow{OP_{cyc(i+1)}}\right|_{\text{if } x_i \le 0 \text{ and } x_{cyc(i+1)} \le 0}$$

$$A_{FHLL} = \frac{1}{2}\left|-Y_{istart\_x}x_{cyc(istart\_x+1)}\right| + \frac{1}{2}\left|x_{iend\_x}Y_{iend\_x}\right| + \frac{1}{2}\sum_{i=1}^{imax}\left|x_i y_{cyc(i+1)} - y_i x_{cyc(i+1)}\right|_{\text{if } x_i \le 0 \text{ and } x_{cyc(i+1)} \le 0}$$

(2.11)

The above equation therefore gives us the area of the $x \le 0$ sub-portion of the base of the inverted pyramid.

Space-time integration of the conservation law over the $x \le 0$ sub-portion of the inverted pyramid in Fig. 2b would give us the *x*-flux in that specific case. A generalization of that concept for any number of wavefronts emanating from the vertex "O" gives

$$\frac{1}{2}\left|Y_{istart\_x} - Y_{iend\_x}\right|\mathbf{F}^* + A_{FHLL}\mathbf{U}^* + \frac{1}{T^2}\left(\text{Area integral over } OP_{cyc(istart\_x+1)}P_{istart\_x;X}\right)$$
$$+\frac{1}{T^2}\left(\text{Area integral over } OP_{iend\_x;X}P_{iend\_x}\right) + \sum_{i=1}^{imax}\frac{1}{T^2}\left(\text{Area integral over } OP_iP_{cyc(i+1)}\right)_{\text{if } x_i \le 0 \text{ and } x_{cyc(i+1)} \le 0} = 0$$

(2.12)

All the terms in the above equation have been made explicit. As a result, the above formula explicitly gives us the subsonic *x*-flux $\mathbf{F}^*$ from the multidimensional HLL Riemann solver. Eqn. (2.12) is one of the most important equations in this paper because it expresses *x*-flux in the strongly interacting subsonic state, $\mathbf{F}^*$, in terms of $\mathbf{U}^*$ and panel integrals taken over some of the sides of our wave model.

We end this sub-Section by making the terms in eqn. (2.12) intuitively obvious. All the terms represent area integrals of our conservation law over the $x \le 0$ portion of our wave model. We use Fig. 2b as an example of a three-sided wave model and the plan view in Fig. 6a as an example of a five-sided wave model. Both figures give us useful perspective which is why we flip back and forth between them as we try to make the terms intuitively obvious. Recall too that the number of sides that a wave model possesses depends on the number of control volumes that come together at the vertex where the multidimensional Riemann problem is being solved. The five terms in eqn. (2.12) have the following simple interpretations:

1) The first term in eqn. (2.12) , $\frac{1}{2}\left|Y_{istart\_x} - Y_{iend\_x}\right|\mathbf{F}^*$, is an integration of our conservation law over the plane $x = 0$ in our wave model. Taking the wave model in Fig. 2b as an example, this is

the integral over ΔOQ_2Q_1. Fig. 6a shows that the $x=0$ plane intersects the side panels labeled "*istart_x*" and "*iend_x*" at *y*-coordinates given by Y_{istart_x} and Y_{iend_x}.

2) The second term in eqn. (2.12), $A_{FHLL}\mathbf{U}^*$, is an integration of our conservation law over the base of our inverted pyramid. Taking the plan view shown for the wave model in Fig. 6a, we can identify A_{FHLL} for a five-sided wave model. For the three-sided wave model shown in Fig. 2b, A_{FHLL} is the area of $\Delta P_3Q_1Q_2$.

3) The third term in eqn. (2.12) is an integration of our conservation law over the incomplete side panel that is labeled "*istart_x*" in Fig. 6a. If we take the wave model in Fig. 2b as an alternative example, this would be the ΔOP_3Q_2 in that figure.

4) The fourth term in eqn. (2.12) is an integration of our conservation law over the incomplete side panel that is labeled "*iend_x*" in Fig. 6a. If we take the wave model in Fig. 2b as an alternative example, this would be the ΔOQ_1P_3 in that figure.

5) The last term in eqn. (2.12) is a sum of panel integrals over all side panels in our wave model that lie entirely to the left of the *y*-axis, i.e. with $x \le 0$. The three-sided wave model in Fig. 2b is too simple and has no such panels. However, the five-sided wave model, shown in plan view in Fig. 6a, has two such panels.

This completes our intuitive description of eqn. (2.12).

II.3.c) Obtaining the Multidimensional HLL y-Flux in the Subsonic limit:

This sub-section follows the previous one very closely. As a result, we do not provide too many detailed explanations. Instead, for the sake of completeness, we just enumerate the important results. For the specific case shown in Fig. 2b, the numerical *y*-flux is obtained by integrating eqn. (2.1) over the sub-portion of the strongly interacting state that has $y \le 0$. Just as Fig. 6a helped us with the *x*-flux, Fig. 6b will help us a lot with the *y*-flux.

Wavefronts that lie entirely below $y=0$ are easily identified because both their end-points in eqn. (2.4) will have *y*-coordinates that are less than or equal to zero. The panel integrals over the side panels of the inverted pyramid are easily obtained for those wavefronts by using eqn. (2.6). Wavefronts which have both end points with *y*-coordinates that are greater than or equal to zero will not contribute to the integrals over the side panels of the inverted pyramid.

For each of the remaining wavefronts in eqn. (2.3), we find the intersection of the plane $y=0$ with those line segments. The point of intersection of $y=0$ with the line segment $P_iP_{cyc(i+1)}$ is given by $P_{i;Y} = \left(X_iT, 0, T\right)$ where $X_i = S_i/\eta_{ix}$. As before, we draw on the convexity of the wave model to find indices "*i=istart_y*" and "*i=iend_y*". In practice, "*istart_y*" is identified by

the condition $\left(y_{istart\_y} > 0\right) and \left(y_{cyc(istart\_y+1)} \le 0\right)$. Likewise, in practice, "*iend_y*" is identified by the condition $\left(y_{iend\_y} \le 0\right) and \left(y_{cyc(iend\_y+1)} > 0\right)$.

After an obvious T^2 scaling, we use the results from Appendix C to integrate over the "*istart_y*" side panel from l_{istart_y} to $l^+_{istart\_y}$. We get:

$$\frac{1}{T^2}\left(\text{Area integral over OP}_{cyc(istart\_y+1)}\text{P}_{istart\_y;Y}\right) =$$

$$\frac{1}{2}\left[\begin{array}{l} \max\left(S^-_{\boldsymbol{\tau} istart\_y} - l_{istart\_y}, 0\right)\left(\mathbf{G}_{\boldsymbol{\eta} istart\_y} - S_{istart\_y}\mathbf{U}_{istart\_y}\right) \\ +\left(S^+_{\boldsymbol{\tau} istart\_y} - \max\left(S^-_{\boldsymbol{\tau} istart\_y}, l_{istart\_y}\right)\right) H\left(S^+_{\boldsymbol{\tau} istart\_y} - l_{istart\_y}\right)\left(\mathbf{G}^*_{\boldsymbol{\eta} istart\_y} - S_{istart\_y}\mathbf{U}^*_{istart\_y}\right) \\ +\left(l^+_{istart\_y} - \max\left(S^+_{\boldsymbol{\tau} start\_y}, l_{istart\_y}\right)\right)\left(\mathbf{G}_{\boldsymbol{\eta} cyc(istart\_y+1)} - S_{istart\_y}\mathbf{U}_{cyc(istart\_y+1)}\right) \end{array}\right] \tag{2.13}$$

As before, remember that the fluxes "$\mathbf{G}_{\boldsymbol{\eta}}$" in the above equation are in the $\boldsymbol{\eta}_{istart\_y}$ direction. We also use the results from Appendix C to integrate over the "*iend_y*" side panel from $l^-_{iend\_y}$ to l_{iend_y}. We get:

$$\frac{1}{T^2}\left(\text{Area integral over OP}_{iend\_y;Y}\text{P}_{iend\_y}\right) =$$

$$\frac{1}{2}\left[\begin{array}{l} \left(\min\left(S^-_{\boldsymbol{\tau} iend\_y}, l_{iend\_y}\right) - l^-_{iend\_y}\right)\left(\mathbf{G}_{\boldsymbol{\eta} iend\_y} - S_{iend\_y}\mathbf{U}_{iend\_y}\right) \\ +\left(\min\left(S^+_{\boldsymbol{\tau} iend\_y}, l_{iend\_y}\right) - S^-_{\boldsymbol{\tau} iend\_y}\right)\text{H}\left(l_{iend\_y} - S^-_{\boldsymbol{\tau} iend\_y}\right)\left(\mathbf{G}^*_{\boldsymbol{\eta} iend\_y} - S_{iend\_y}\mathbf{U}^*_{iend\_y}\right) \\ +\max\left(l_{iend\_y} - S^+_{\boldsymbol{\tau} iend\_y}, 0\right)\left(\mathbf{G}_{\boldsymbol{\eta} cyc(iend\_y+1)} - S_{iend\_y}\mathbf{U}_{cyc(iend\_y+1)}\right) \end{array}\right] \tag{2.14}$$

As before, remember that the fluxes "$\mathbf{G}_{\boldsymbol{\eta}}$" in the above equation are in the $\boldsymbol{\eta}_{iend\_y}$ direction. The rest of the area integrals over the side panels are exactly like eqn. (2.6).

The area that is contained within the thick line in Fig. 6b is called A_{GHLL}. After a T^2 scaling, it is given by a series of cross products:

$$\text{A}_{GHLL}T^2 = \frac{1}{2}\left|\overrightarrow{\text{OP}_{istart\_y;Y}} \times \overrightarrow{\text{OP}_{cyc(istart\_y+1)}}\right| + \frac{1}{2}\left|\overrightarrow{\text{OP}_{iend\_y}} \times \overrightarrow{\text{OP}_{iend\_y;Y}}\right| + \frac{1}{2}\sum_{i=1}^{imax}\left|\overrightarrow{\text{OP}_i} \times \overrightarrow{\text{OP}_{cyc(i+1)}}\right|_{\text{if } y_i \le 0 \text{ and } y_{cyc(i+1)} \le 0}$$

$$\text{A}_{GHLL} = \frac{1}{2}\left|X_{istart\_y} y_{cyc(istart\_y+1)}\right| + \frac{1}{2}\left|-y_{iend\_y} X_{iend\_y}\right| + \frac{1}{2}\sum_{i=1}^{imax}\left|x_i y_{cyc(i+1)} - y_i x_{cyc(i+1)}\right|_{\text{if } y_i \le 0 \text{ and } y_{cyc(i+1)} \le 0} \tag{2.15}$$

Space-time integration of the conservation law over that part of the inverted pyramid in Fig. 2b which has $y \le 0$ gives

$$\frac{1}{2}\left|X_{iend\_y}-X_{istart\_y}\right|\mathbf{G}^{*}+\mathrm{A}_{GHLL}\mathbf{U}^{*}+\frac{1}{T^{2}}\left(\text{Area integral over OP}_{cyc(istart\_y+1)}\text{P}_{istart\_y;Y}\right)$$
$$+\frac{1}{T^{2}}\left(\text{Area integral over OP}_{iend\_y;Y}\text{P}_{iend\_y}\right)+\sum_{i=1}^{imax}\frac{1}{T^{2}}\left(\text{Area integral over OP}_{i}\text{P}_{cyc(i+1)}\right)_{\text{if } y_i\le 0 \text{ and } y_{cyc(i+1)}\le 0}=0$$

(2.16)

All the terms in the above equation have been made explicit. As a result, the above formula explicitly gives us the subsonic *y*-flux $\mathbf{G}^{*}$ from the multidimensional Riemann solver. Eqn. (2.16) is one of the most important equations in this paper because it expresses *y*-flux in the strongly interacting subsonic state, $\mathbf{G}^{*}$, in terms of $\mathbf{U}^{*}$ and panel integrals taken over some of the sides of our wave model.

Using the plan view of the five-sided wave model shown in Fig. 6b, we can give an intuitive introduction to eqn. (2.16). The five terms in eqn. (2.16) have the following simple interpretations:

1) The first term in eqn. (2.16) , $\frac{1}{2}\left|X_{iend\_y}-X_{istart\_y}\right|\mathbf{G}^{*}$, is an integration of our conservation law over the plane $y=0$ in our wave model. Fig. 6b shows that the $y=0$ plane intersects the side panels labeled "*istart _ y*" and "*iend _ y*" at *x*-coordinates given by X_{istart_y} and X_{iend_y}.

2) The second term in eqn. (2.16), $\mathrm{A}_{GHLL}\mathbf{U}^{*}$, is an integration of our conservation law over the base of our inverted pyramid. Taking the plan view shown for the wave model in Fig. 6b, we can identify A_{GHLL} for a five-sided wave model.

3) The third term in eqn. (2.16) is an integration of our conservation law over the incomplete side panel that is labeled "*istart _ y*" in Fig. 6b.

4) The fourth term in eqn. (2.16) is an integration of our conservation law over the incomplete side panel that is labeled "*iend _ y*" in Fig. 6b.

5) The last term in eqn. (2.16) is a sum of panel integrals over all side panels in our wave model that lie entirely below the *x*-axis, i.e. with $y\le 0$. The five-sided wave model, shown in plan view in Fig. 6b, has one such panel.

This completes our intuitive description of eqn. (2.16).

II.3.d) Obtaining the Numerical Fluxes in the Supersonic limit

When the second variant of the wave model is used, see Appendix B, there is no need to consider a separate supersonic limit. The rest of this sub-section pertains to situations when the first variant of the wave model in Appendix B is used.

The strongly interacting state $\mathbf{U}^*$ is always obtained using the methods in Sub-section II.3.a. When the problem is supersonic, the fluxes are not obtained from Sub-sections II.3.b and II.3.c. In the supersonic case, it helps if one has some multidimensional proxy for the upwinding direction. This is easily obtained from $\mathbf{U}^*$ by evaluating the associated density and *x*- and *y*-velocities as

$$\rho^* = \left(\mathbf{U}^*\right)_1 \quad ; \quad S^*_{Mx} = \frac{\left(\mathbf{U}^*\right)_2}{\rho^*} \quad ; \quad S^*_{My} = \frac{\left(\mathbf{U}^*\right)_3}{\rho^*} \tag{2.17}$$

We then form the vector $\mathbf{A} = -S^*_{Mx}\mathbf{x} - S^*_{My}\mathbf{y}$ which forms a good physics-based proxy for the direction of upwinding in the supersonic case. We then obtain the index "*i=iupwind*" by polling the principal directions of the mesh for the condition

$$\left(\eta_{ix}S^*_{My} - \eta_{iy}S^*_{Mx}\right)\left(\eta_{cyc(i+1)x}S^*_{My} - \eta_{cyc(i+1)y}S^*_{Mx}\right) \le 0 \tag{2.18}$$

With "*iupwind*" in hand, we evaluate the angle $\xi_{iupwind}$ between $\mathbf{A}$ and $\boldsymbol{\eta}_{iupwind}$. We also evaluate $\xi_{cyc(iupwind+1)}$ between $\mathbf{A}$ and $\boldsymbol{\eta}_{cyc(iupwind+1)}$. In practice, this is achieved by defining

$$\sin\xi_{iupwind} = \left|\boldsymbol{\eta}_{iupwind} \times \mathbf{A}\right| \Big/ \left|\mathbf{A}\right| \quad ; \quad \sin\xi_{cyc(iupwind+1)} = \left|\mathbf{A} \times \boldsymbol{\eta}_{cyc(iupwind+1)}\right| \Big/ \left|\mathbf{A}\right| \tag{2.19}$$

We then build the angle-weighted weights

$$\begin{aligned} w_{iupwind} &= \frac{1}{\left(\sin^2\xi_{iupwind} + \varepsilon\right)} \quad ; \quad w_{cyc(iupwind+1)} = \frac{1}{\left(\sin^2\xi_{cyc(iupwind+1)} + \varepsilon\right)} \quad ; \\ \overline{w}_{iupwind} &= \frac{w_{iupwind}}{w_{iupwind} + w_{cyc(iupwind+1)}} \quad ; \quad \overline{w}_{cyc(iupwind+1)} = \frac{w_{cyc(iupwind+1)}}{w_{iupwind} + w_{cyc(iupwind+1)}} \end{aligned} \tag{2.20}$$

We took ε to be some very small number like 10^{-8}. Let $\mathbf{F}_i^{HLL1D}$ and $\mathbf{G}_i^{HLL1D}$ be the *x*- and *y*-fluxes coming from the one-dimensional HLL Riemann solver at the i^{th} boundary. The final, supersonically upwinded *x*- and *y*-fluxes are given by

$$\begin{aligned} \mathbf{F}^* &= \overline{w}_{iupwind}\ \mathbf{F}^{HLL1D}_{iupwind} + \overline{w}_{cyc(iupwind+1)}\ \mathbf{F}^{HLL1D}_{cyc(iupwind+1)} \quad ; \\ \mathbf{G}^* &= \overline{w}_{iupwind}\ \mathbf{G}^{HLL1D}_{iupwind} + \overline{w}_{cyc(iupwind+1)}\ \mathbf{G}^{HLL1D}_{cyc(iupwind+1)} \end{aligned} \tag{2.21}$$

The numerical fluxes that we use in the supersonic limit are given by the above equation.

In the supersonic limit, the two-dimensional HLL and HLLC Riemann solvers should yield the same numerical fluxes. As a result, the formulation for the numerical fluxes in the

supersonic limit that is presented here can indeed be naturally extended to the multidimensional HLLC Riemann solver.

The multidimensional HLL Riemann solver described in this Section is implemented on a computer by following the logic of the Sub-sections, along with their respective equations, in the very same order that they are written here. Please do pay attention to the discussions in Balsara [5] because the Riemann solver is indeed implemented at the vertices rather than at the zone boundaries of a mesh.

III) Multidimensional HLLC Riemann Solver

From Schulz-Rinne, Collins and Glaz [52] we see that contact discontinuities that are oriented in any possible direction on the computational mesh constitute one of the persistent flow structures in the strong-interaction region of many multi-dimensional Riemann problems. For that reason, we want our multi-dimensional HLLC Riemann solver to resolve contact discontinuities in the region of strong interaction. Figs. 7a and 7b depict the changes in the wave model when an intermediate wave, corresponding to the contact discontinuity, is introduced in the original strongly-interacting state. At each stage in the discussion, these figures should be compared to Figs. 2a and 2b to appreciate the differences between the multidimensional HLL and HLLC Riemann solvers. Our present formulation of the multidimensional HLLC Riemann solver for unstructured meshes draws on original insights that were developed in Balsara [5] for structured meshes.

Between each pair of constant states in Fig. 1, we now solve a one-dimensional HLLC Riemann solver. Compare Fig. 7a to Fig. 2a to appreciate the change that takes place in the wave model. The notation describing Fig. 7a is quite similar to the one describing Fig. 2a. As a result, we only describe the wave model in the face $\mathrm{P_1'P_2'P_2P_1}$ of Fig. 7a. The two extremal waves in that face, i.e. $S_{\tau 1}^{-}$ and $S_{\tau 1}^{+}$, now have a contact discontinuity propagating between them with speed $S_{\tau 1}^{M}$. Instead of the intermediate state $\mathbf{U}_1^*$ from Fig. 2a, we have two states, $\mathbf{U}_1^{*-}$ and $\mathbf{U}_1^{*+}$, between the extremal waves that are separated by the contact discontinuity. As before, by using a few steps, we can obtain the *x*- and *y*-fluxes $\mathbf{F}_1^{*-}$ and $\mathbf{G}_1^{*-}$ corresponding to $\mathbf{U}_1^{*-}$. Similarly, we can obtain the *x*- and *y*-fluxes $\mathbf{F}_1^{*+}$ and $\mathbf{G}_1^{*+}$ corresponding to $\mathbf{U}_1^{*+}$. Similar simplifications can be made for the fluxes in other directions.

The original strongly-interacting state, shown in Fig. 2, is now sub-divided into two strongly-interacting states, $\mathbf{U}_{C1}^*$ and $\mathbf{U}_{C2}^*$, that are separated by a contact discontinuity, as shown in Fig. 7. The state $\mathbf{U}_{C1}^*$ is of greater interest because it straddles the time axis. In keeping with the spirit of a multidimensional approach, the discontinuity can move in any direction in the *x*-*y* plane with *x*- and *y*-speeds given by S_{Mx}^* and S_{My}^*, see Fig. 7 and eqn. (2.17). The velocity with which the contact discontinuity propagates, as well as the extent of the density jump across it, is set by the Riemann solver.

The physics of contact discontinuities tells us that a discontinuity can be passively advected by the velocity vector. As a result, the contact discontinuity can make any angle to the velocity vector and we hope that Fig. 7 also illustrates this. The angle that the contact discontinuity makes relative to the mesh can only be determined by evaluating the gradient of the density from the three or more zones that come together at the edge "O". As a result, we assume that the unit normal vector to the contact discontinuity, given by $\boldsymbol{\sigma} \equiv \sigma_x \mathbf{x} + \sigma_y \mathbf{y}$, is also provided to the Riemann solver. The locus of the contact discontinuity at time *T* in Fig. 7 is given by

$$y = S_{My}^* T \; - \; \frac{\sigma_x}{\sigma_y} \left(x - S_{Mx}^* T \right) \tag{3.1}$$

The contact discontinuity intersects the triangle $\Delta P_1 P_2 P_3$ at the points Q_3 and Q_4, as shown in Fig. 7. Eqns. (2.3) and (3.1) can be used to obtain the locations of points Q_3 and Q_4 in space and time. Notice that eqn. (3.1) of this paper matches with eqn. (21) from Balsara [5] for the multidimensional HLLC Riemann solver on structured meshes.

Every HLLC Riemann solver is built on top of an HLL Riemann solver and the present one is no exception. Thus we first use eqns. (2.8), (2.12) and (2.16) to obtain $\mathbf{U}^*$, $\mathbf{F}^*$, and $\mathbf{G}^*$. We follow the suggestion of Toro, Spruce & Speares [51] and Batten *et al.* [15] and operationally define the pressure in the strongly-interacting state. For the Euler system we get

$$P^* = \frac{1}{2} \left\{ \left[\left(\mathbf{F}^* \right)_2 - \rho^* \left(S_{Mx}^* \right)^2 \right] + \left[\left(\mathbf{G}^* \right)_3 - \rho^* \left(S_{My}^* \right)^2 \right] \right\} \tag{3.2}$$

The physics of contact discontinuities tells us that P^*, S_{Mx}^* and S_{My}^* remain constant across the contact discontinuity with the result that the pressure and velocities in the states $\mathbf{U}_{C1}^*$ and $\mathbf{U}_{C2}^*$ are fully specified. Now that P^*, S_{Mx}^* and S_{My}^* are available for $\mathbf{U}_{C1}^*$ and $\mathbf{U}_{C2}^*$, we can formally write down the expressions for the numerical *x*- and *y*-fluxes in the strongly-interacting state $\mathbf{U}_{C1}^*$ of interest as

$$\mathbf{F}_{C1}^* = S_{Mx}^* \, \mathbf{U}_{C1}^* + \left(0, P^*, 0, 0, P^* S_{Mx}^* \right)^T \quad ; \quad \mathbf{G}_{C1}^* = S_{My}^* \, \mathbf{U}_{C1}^* + \left(0, 0, P^*, 0, P^* S_{My}^* \right)^T \tag{3.3}$$

In other words, the hydrodynamical fluxes in the *x*- and *y*-directions are linear combinations of $\mathbf{U}_{C1}^*$ along with the addition of a constant vector that is already known. Such a linearization plays a crucial step in the derivation of the multidimensional HLLC Riemann solver. The above fluxes would be completely specified if $\mathbf{U}_{C1}^*$ were specified. For that reason, from the next paragraph onwards, we embark on an implementable strategy for obtaining $\mathbf{U}_{C1}^*$. Notice that eqns. (2.17), (3.2) and (3.3) of this paper are the exact analogues of eqns. (18), (19) and (20) from Balsara [5] for the multidimensional HLLC Riemann solver on structured meshes.

For the specific case shown in Fig. 7b, the state $\mathbf{U}^*_{C1}$ for the multidimensional HLLC Riemann solver can be obtained by integrating the conservation law over the inverted pyramid with quadrilateral $P_1Q_3Q_4P_3$ as the base and "O" as the apex. Figs. 6c and 6d show us a couple of different top views of the wave model used for obtaining $\mathbf{U}^*_{C1}$ in the general case when more than three zones come together at the vertex "O". As in the previous Section, we split the overall task of carrying out the space-time integration into a sequence of sub-tasks. They are:

1) Geometric Analysis: We wish to obtain an automated method for identifying "*i=istart_c*" and "*i=iend_c*" in Figs. 6c and 6d. The method should also distinguish points in space-time that lie on the same side of the contact as $\mathbf{U}^*_{C1}$ from points that lie on the other side, i.e. the side occupied by $\mathbf{U}^*_{C2}$. We are not much interested in the latter side though.

2) Obtain the Areas: Notice that we will use Gauss law to convert volume integrals in Fig. 7b to area integrals over surfaces that contain the state $\mathbf{U}^*_{C1}$. Some of those areas are non-trivial and we provide the steps for evaluating them.

3) Integrate the Conservation Law over the Areas to Obtain the States and Fluxes: Sub-section II.3 has already shown us how the area integrals can be carried out over the side panels in Fig. 2b. The wave model in the side panels of Fig. 7b is a little more complex owing to the introduction of the contact discontinuity in the one-dimensional HLLC Riemann solvers. Consequently, we provide explicit expressions for the integrals. The integrals are subsequently used to obtain the state $\mathbf{U}^*_{C1}$ and the associated *x*- and *y*-fluxes for the multidimensional HLLC Riemann solver in the subsonic limit. The state $\mathbf{U}^*_{C2}$ does not overlie the time axis; however, it too can be obtained. There are two alternative methods for obtaining the states and fluxes. As a result, we provide two sub-sections. It is trivial to extend the results of Sub-section II.3.d to the supersonic HLLC case and we do not repeat it here.

The next four Sub-sections detail the above three tasks. In these Sub-sections we also show the correspondence between the two-dimensional HLLC Riemann solvers on structured and unstructured meshes in order to facilitate understanding. The formulae in this Section follow the same sequence that one should use in making a computer implementation of the two-dimensional HLLC Riemann solver. This should simplify the process of making a computer implementation.

III.1) Geometric Analysis:

The wavefront associated with the contact discontinuity is given by eqn. (3.1). Its intersection with the i^{th} wavefront in eqn. (2.3) is given by the point $C_i = \left(X_{ic}T, Y_{ic}T, T\right)$ where we have

$$X_{ic} = \frac{\left(\sigma_x S^*_{Mx} + \sigma_y S^*_{My}\right)\eta_{iy} - \sigma_y S_i}{\sigma_x \eta_{iy} - \sigma_y \eta_{ix}} \quad ; \quad Y_{ic} = -\frac{\left(\sigma_x S^*_{Mx} + \sigma_y S^*_{My}\right)\eta_{ix} - \sigma_x S_i}{\sigma_x \eta_{iy} - \sigma_y \eta_{ix}} \tag{3.4}$$

Please note that X_{ic} and Y_{ic} have units of speed. Again, we scan the convex-hulled wave model to find the two unique wavefronts that make an internal intersection with the contact discontinuity. Let those wavefronts be given by indices "*i=istart_c*" and "*i=iend_c*" in Figs. 6c and 6d. Notice from Figs. 6c and 6d that the arrangement of side panels to be integrated over is such that we go from "*i=istart_c*" to "*i=iend_c*" in the counter-clockwise direction. The state $\mathbf{U}^*_{C1}$, by convention, straddles the time axis. In the next paragraph we will provide an automatic process for picking out the state $\mathbf{U}^*_{C1}$.

Recall that U^*_{C1} overlies the origin "O" so that the origin can be used to provide a sense of sidedness. Any point (x, y, t) is on the same side as the origin "O" in Figs. 6c and 6d if $\sigma_x\left(x - S^*_{Mx}\right) + \sigma_y\left(y - S^*_{My}\right)$ has the same sign as $-\sigma_x S^*_{Mx} - \sigma_y S^*_{My}$. Thus we say that a general point (x, y) lies on the same side as the origin "O" if $\left[\sigma_x\left(x - S^*_{Mx}\right) + \sigma_y\left(y - S^*_{My}\right)\right]\left(\sigma_x S^*_{Mx} + \sigma_y S^*_{My}\right) \le 0$. We want an automatic process for picking out side panels in Fig. 7b that are on the same side as the state U^*_{C1}. Consequently, we say that a wavefront $\mathrm{P}_i\mathrm{P}_{cyclic(i+1)}$ with end points (x_i, y_i) and $\left(x_{cyc(i+1)}, y_{cyc(i+1)}\right)$ lies entirely on the desired side if we have the following condition

$$\left\{\left[\sigma_x\left(x_i - S^*_{Mx}\right) + \sigma_y\left(y_i - S^*_{My}\right)\right]\left(\sigma_x S^*_{Mx} + \sigma_y S^*_{My}\right) \le 0\right\} \textit{ and } \left\{\left[\sigma_x\left(x_{cyc(i+1)} - S^*_{Mx}\right) + \sigma_y\left(y_{cyc(i+1)} - S^*_{My}\right)\right]\left(\sigma_x S^*_{Mx} + \sigma_y S^*_{My}\right) \le 0\right\} \tag{3.5}$$

We can identify "*i=istart_c*" by the condition

$$\left\{\left[\sigma_x\left(x_{istart\_c} - S^*_{Mx}\right) + \sigma_y\left(y_{istart\_c} - S^*_{My}\right)\right]\left(\sigma_x S^*_{Mx} + \sigma_y S^*_{My}\right) > 0\right\} \textit{ and } \left\{\left[\sigma_x\left(x_{cyc(istart\_c+1)} - S^*_{Mx}\right) + \sigma_y\left(y_{cyc(istart\_c+1)} - S^*_{My}\right)\right]\left(\sigma_x S^*_{Mx} + \sigma_y S^*_{My}\right) \le 0\right\} \tag{3.6}$$

We can identify "*i=iend_c*" by the condition

$$\left\{\left[\sigma_x\left(x_{iend\_c} - S^*_{Mx}\right) + \sigma_y\left(y_{iend\_c} - S^*_{My}\right)\right]\left(\sigma_x S^*_{Mx} + \sigma_y S^*_{My}\right) \le 0\right\} \textit{ and } \left\{\left[\sigma_x\left(x_{cyc(iend\_c+1)} - S^*_{Mx}\right) + \sigma_y\left(y_{cyc(iend\_c+1)} - S^*_{My}\right)\right]\left(\sigma_x S^*_{Mx} + \sigma_y S^*_{My}\right) > 0\right\} \tag{3.7}$$

This tells us how to identify all the side panels in Fig. 7b that give us non-trivial contributions to the integrals. If the term $\left(\sigma_x S^*_{Mx} + \sigma_y S^*_{My}\right)$ evaluates to zero, it means that the contact discontinuity passes through the origin. In that case, the above-mentioned term can be dropped from eqns. (3.5) to (3.7) because either of the two states in Fig. 7b can play the role of $\mathbf{U}^*_{C1}$.

III.2) Obtain the Areas:

The area along the time axis that covers the state U^*_{C1} is denoted by A_{UHLLC} . It is bounded by the thick lines in Figs. 6c and 6d. After a T^2 scaling, it is given by a series of cross products in the two-dimensional *xy*-planes shown in Figs. 6c and 6d as:

$$\begin{aligned}
\mathrm{A}_{UHLLC}T^2 &= \frac{1}{2}\left|\overrightarrow{\mathrm{OC}_{iend_c}}\times\overrightarrow{\mathrm{OC}_{istart_c}}\right| + \frac{1}{2}\left|\overrightarrow{\mathrm{OC}_{istart_c}}\times\overrightarrow{\mathrm{OP}_{cyc(istart_c+1)}}\right| + \frac{1}{2}\left|\overrightarrow{\mathrm{OP}_{iend_c}}\times\overrightarrow{\mathrm{OC}_{iend_c}}\right| \\
&\quad + \frac{1}{2}\sum_{i=1}^{imax}\left|\overrightarrow{\mathrm{OP}_i}\times\overrightarrow{\mathrm{OP}_{cyc(i+1)}}\right|_{\mathrm{if}\left[\sigma_x\left(x-S^*_{Mx}\right)+\sigma_y\left(y-S^*_{My}\right)\right]\left(\sigma_x S^*_{Mx}+\sigma_y S^*_{My}\right)\le 0} \\
\mathrm{A}_{UHLLC} &= \frac{1}{2}\left|X_{iend_c}Y_{istart_c}-Y_{iend_c}X_{istart_c}\right| + \frac{1}{2}\left|X_{istart_c}y_{cyc(istart_c+1)}-Y_{istart_c}x_{cyc(istart_c+1)}\right| \\
&\quad + \frac{1}{2}\left|x_{iend_c}Y_{iend_c}-y_{iend_c}X_{iend_c}\right| \\
&\quad + \frac{1}{2}\sum_{i=1}^{imax}\left|x_i y_{cyc(i+1)}-y_i x_{cyc(i+1)}\right|_{\mathrm{if}\left[\sigma_x\left(x-S^*_{Mx}\right)+\sigma_y\left(y-S^*_{My}\right)\right]\left(\sigma_x S^*_{Mx}+\sigma_y S^*_{My}\right)\le 0}
\end{aligned} \tag{3.8}$$

In order to obtain the outward-pointing area associated with $\Delta\mathrm{OQ}_4\mathrm{Q}_3$ in Fig. 7b, we now define

$$\begin{aligned}
N_x^{C1C2}\mathbf{x}+N_y^{C1C2}\mathbf{y}+N_t^{C1C2}\mathbf{t} &\equiv \frac{1}{2T^2}\overrightarrow{\mathrm{OC}_{istart_c}}\times\overrightarrow{\mathrm{OC}_{iend_c}} \\
&= \frac{1}{2}\left(\mathrm{X}_{istart_c}\mathbf{x}+\mathrm{Y}_{istart_c}\mathbf{y}+\mathbf{t}\right)\times\left(\mathrm{X}_{iend_c}\mathbf{x}+\mathrm{Y}_{iend_c}\mathbf{y}+\mathbf{t}\right)
\end{aligned} \tag{3.9}$$

Our choice of "*i=istart_c*" and "*i=iend_c*" is such that the above equation will always give us an outward pointing normal vector in three dimensional space-time. It is also easy to demonstrate that $N_x^{C1C2}\sigma_y - N_y^{C1C2}\sigma_x = 0$, which proves that the vector $N_x^{C1C2}\mathbf{x}+N_y^{C1C2}\mathbf{y}$ is aligned with the normal to the contact discontinuity $\sigma_x\mathbf{x}+\sigma_y\mathbf{y}$. This result is independent of the velocity of the contact discontinuity $S^*_{Mx}\mathbf{x}+S^*_{My}\mathbf{y}$. Notice too that the areas in eqns. (3.8) and (3.9) of this paper are the analogues of the areas in eqns. (23) and (24) from Balsara [5] for the multidimensional HLLC Riemann solver on structured meshes.

III.3) Integrate the Conservation Law over the Areas to Obtain the States and Fluxes:

The area integrals over the side panels in Fig. 7b now get upgraded from eqn. (2.6) to

$$\frac{1}{T^2}\Big(\text{Area integral over OP}_i\text{P}_{cyc(i+1)}\Big)=$$
$$\frac{1}{2}\left[\begin{array}{l}\left(S_{\tau i}^{-}-l_i^{-}\right)\left(\mathbf{G}_{\eta i}-S_i\mathbf{U}_i\right)+\left(S_{\tau i}^{M}-S_{\tau i}^{-}\right)\left(\mathbf{G}_{\eta i}^{*-}-S_i\mathbf{U}_i^{*-}\right)\\+\left(S_{\tau i}^{+}-S_{\tau i}^{M}\right)\left(\mathbf{G}_{\eta i}^{*+}-S_i\mathbf{U}_i^{*+}\right)+\left(l_i^{+}-S_{\tau i}^{+}\right)\left(\mathbf{G}_{\eta cyc(i+1)}-S_i\mathbf{U}_{cyc(i+1)}\right)\end{array}\right] \tag{3.10}$$

The fluxes "$\mathbf{G}_{\eta}$" in the above equation are in the $\boldsymbol{\eta}_i$ direction. The results from Appendix C have been used to obtain the previous equation. The other two additional area integrals, which extend only over part of a side panel in Fig. 7b, now need to be evaluated. Let l_{istart_c} be the local coordinate of point C_{istart_c} in the "*istart_c*" panel. The area integral over the "*istart_c*" side panel from l_{istart_c} to $l_{istart\_c}^{+}$ is given by

$$\frac{1}{T^2}\Big(\text{Area integral over OC}_{istart\_c}\text{P}_{cyc(istart\_c+1)}\Big)=$$
$$\frac{1}{2}\left[\begin{array}{l}\max\left(S_{\tau istart\_c}^{-}-l_{istart\_c},0\right)\left(\mathbf{G}_{\eta istart\_c}-S_{istart\_c}\mathbf{U}_{istart\_c}\right)\\+\left(S_{\tau istart\_c}^{M}-\max\left(S_{\tau istart\_c}^{-},l_{istart\_c}\right)\right)\text{H}\left(S_{\tau istart\_c}^{M}-l_{istart\_c}\right)\left(\mathbf{G}_{\eta istart\_c}^{*-}-S_{istart\_c}\mathbf{U}_{istart\_c}^{*-}\right)\\+\left(S_{\tau istart\_c}^{+}-\max\left(S_{\tau istart\_c}^{M},l_{istart\_c}\right)\right)\text{H}\left(S_{\tau istart\_c}^{+}-l_{istart\_c}\right)\left(\mathbf{G}_{\eta istart\_c}^{*+}-S_{istart\_c}\mathbf{U}_{istart\_c}^{*+}\right)\\+\left(l_{istart\_c}^{+}-\max\left(S_{\tau istart\_c}^{+},l_{istart\_c}\right)\right)\left(\mathbf{G}_{\eta cyc(istart\_c+1)}-S_{istart\_c}\mathbf{U}_{cyc(istart\_c+1)}\right)\end{array}\right] \tag{3.11}$$

The fluxes "$\mathbf{G}_{\eta}$" in the above equation are in the $\boldsymbol{\eta}_{istart\_c}$ direction. The results from Appendix C have been used to obtain the previous equation. Let l_{iend_c} be the local coordinate of point C_{iend_c} in the "*iend_c*" panel. The area integral over the "*iend_c*" side panel from $l_{iend\_c}^{-}$ to l_{iend_c} is given by

$$\frac{1}{T^2}\Big(\text{Area integral over OP}_{iend\_c}\text{C}_{iend\_c}\Big)=$$
$$\frac{1}{2}\left[\begin{array}{l}\left(\min\left(S_{\tau iend\_c}^{-},l_{iend\_c}\right)-l_{iend\_c}^{-}\right)\left(\mathbf{G}_{\eta iend\_c}-S_{iend\_c}\mathbf{U}_{iend\_c}\right)\\+\left(\min\left(S_{\tau iend\_c}^{M},l_{iend\_c}\right)-S_{\tau iend\_c}^{-}\right)\text{H}\left(l_{iend\_c}-S_{\tau iend\_c}^{-}\right)\left(\mathbf{G}_{\eta iend\_c}^{*-}-S_{iend\_c}\mathbf{U}_{iend\_c}^{*-}\right)\\+\left(\min\left(S_{\tau iend\_c}^{+},l_{iend\_c}\right)-S_{\tau iend\_c}^{M}\right)\text{H}\left(l_{iend\_c}-S_{\tau iend\_c}^{M}\right)\left(\mathbf{G}_{\eta iend\_c}^{*+}-S_{iend\_c}\mathbf{U}_{iend\_c}^{*+}\right)\\+\max\left(l_{iend\_c}-S_{\tau iend\_c}^{+},0\right)\left(\mathbf{G}_{\eta cyc(iend\_c+1)}-S_{iend\_c}\mathbf{U}_{cyc(iend\_c+1)}\right)\end{array}\right] \tag{3.12}$$

The fluxes "$\mathbf{G}_{\eta}$" in the above equation are in the $\boldsymbol{\eta}_{iend\_c}$ direction. The results from Appendix C have been used to obtain the previous equation. Apart from eqns. (3.11) and (3.12), the rest of the area integrals over the side panels are exactly like the generic one above, i.e. as given in eqn. (3.10). Notice too that the panel integral in eqn. (3.12) of this paper is analogous to the panel integral in eqn. (26) from Balsara [5] for the multidimensional HLLC Riemann solver on

structured meshes. Likewise, the panel integrals in eqns. (3.10) and (3.11) are analogous to the two panel integrals in Appendix B of Balsara [5].

The analogue of eqn. (2.8) for $\mathbf{U}^*_{C1}$ is given by

$$\begin{aligned}
&\mathrm{A}_{UHLLC}\mathbf{U}^*_{C1} + \left(N_x^{C1C2}\mathbf{x} + N_y^{C1C2}\mathbf{y} + N_t^{C1C2}\mathbf{t}\right)\bullet\left(\mathbf{F}^*_{C1}\mathbf{x} + \mathbf{G}^*_{C1}\mathbf{y} + \mathbf{U}^*_{C1}\mathbf{t}\right) \\
&\quad + \sum_{i=1}^{imax}\frac{1}{T^2}\Big(\text{Area integral over } \mathrm{OP}_i\mathrm{P}_{cyc(i+1)}\Big)_{\text{if}\left[\sigma_x\left(x-S^*_{Mx}\right)+\sigma_y\left(y-S^*_{My}\right)\right]\left(\sigma_x S^*_{Mx}+\sigma_y S^*_{My}\right)\le 0} \\
&\quad + \frac{1}{T^2}\Big(\text{Area integral over } \mathrm{OC}_{istart_c}\mathrm{P}_{cyc(istart_c+1)}\Big) + \frac{1}{T^2}\Big(\text{Area integral over } \mathrm{OP}_{iend_c}\mathrm{C}_{iend_c}\Big) = 0
\end{aligned} \tag{3.13}$$

The fluxes $\mathbf{F}^*_{C1}$ and $\mathbf{G}^*_{C1}$ from eqn. (3.3) can be incorporated into the above equation to get

$$\begin{aligned}
&\left(\mathrm{A}_{UHLLC} + N_t^{C1C2} + N_x^{C1C2}S^*_{Mx} + N_y^{C1C2}S^*_{My}\right)\mathbf{U}^*_{C1} = \\
&\qquad - N_x^{C1C2}\left(0, P^*, 0, 0, P^*S^*_{Mx}\right)^T - N_y^{C1C2}\left(0, 0, P^*, 0, P^*S^*_{My}\right)^T \\
&\qquad - \sum_{i=1}^{imax}\frac{1}{T^2}\Big(\text{Area integral over } \mathrm{OP}_i\mathrm{P}_{cyc(i+1)}\Big)_{\text{if}\left[\sigma_x\left(x-S^*_{Mx}\right)+\sigma_y\left(y-S^*_{My}\right)\right]\left(\sigma_x S^*_{Mx}+\sigma_y S^*_{My}\right)\le 0} \\
&\qquad - \frac{1}{T^2}\Big(\text{Area integral over } \mathrm{OC}_{istart_c}\mathrm{P}_{cyc(istart_c+1)}\Big) \\
&\qquad - \frac{1}{T^2}\Big(\text{Area integral over } \mathrm{OP}_{iend_c}\mathrm{C}_{iend_c}\Big)
\end{aligned} \tag{3.14}$$

The above equation, along with eqn. (3.3), gives us the state $\mathbf{U}^*_{C1}$ and fluxes $\mathbf{F}^*_{C1}$ and $\mathbf{G}^*_{C1}$ in the strongly interacting state for the multidimensional HLLC Riemann solver. Once $\mathbf{U}^*_{C1}$ is obtained, eqn. (3.3) gives us the fluxes $\mathbf{F}^*_{C1}$ and $\mathbf{G}^*_{C1}$. Eqn. (3.14) is one of the most important equations in this paper because it expresses the subsonic state $\mathbf{U}^*_{C1}$ in terms of expressions that can be simply evaluated.

Since eqn. (3.14) is a direct consequence of applying the linearization in eqn. (3.3) to eqn. (3.13), it is valuable to explain the terms in eqn. (3.13) intuitively. Using the five-sided wave model that is shown in plan view in Fig. 6c as a visual tool, we make the five terms in eqn. (3.13) intuitively obvious. The three-sided wave model in Fig. 7b is also very useful and will be used as an alternate example when appropriate. The five terms in eqn. (3.13) are all integrals of the conservation laws over surfaces that bound sub-portions of our multidimensional wave model. They are interpreted as follows:

1) The first term, $\mathrm{A}_{UHLLC}\mathbf{U}^*_{C1}$, in eqn. (3.13) is an integration of our conservation law over the base of the inverted pyramid that we are considering. The contact discontinuity is shown by the thick dashed line in Fig. 6c which also shows the area A_{UHLLC} occupied by the state $\mathbf{U}^*_{C1}$ which overlies the time axis. Taking Fig. 7b as our alternate example, the base of the inverted pyramid

is given by the quadrangle $P_1Q_3Q_4P_3$, with the origin "O" forming the vertex of the same pyramid. For Fig. 7b, the area A_{UHLLC} would be the area of quadrangle $P_1Q_3Q_4P_3$.

2) The second term, $\left(N_x^{C1C2}\mathbf{x}+N_y^{C1C2}\mathbf{y}+N_t^{C1C2}\mathbf{t}\right)\bullet\left(\mathbf{F}_{C1}^{*}\mathbf{x}+\mathbf{G}_{C1}^{*}\mathbf{y}+\mathbf{U}_{C1}^{*}\mathbf{t}\right)$, in eqn. (3.13) is an integration of our conservation law over the triangle formed by the origin and the two endpoints of the contact discontinuity shown in Fig. 6c. From Fig. 6c we see that the contact discontinuity intersects the two side panels in our wave model labeled by "$istart\_c$" and "$iend\_c$". Let C_{istart_c} and C_{iend_c} be the endpoints of the contact discontinuity in the side panels labeled by "$istart\_c$" and "$iend\_c$" respectively. The area vector $N_x^{C1C2}\mathbf{x}+N_y^{C1C2}\mathbf{y}+N_t^{C1C2}\mathbf{t}$ represents the area of $\Delta OC_{istart\_c}C_{iend\_c}$. The dot product gives the integration of the conservation law over that area. Taking Fig. 7b as an alternative example, the area integral we seek extends over ΔOQ_4Q_3 .

3) The third term in eqn. (3.13) is a summation of panel integrals over all the side panels in our wave model that cover the state $\mathbf{U}_{C1}^{*}$. Moreover, these side panels lie entirely on one side of the contact discontinuity. This is ensured by the selection criterion $\left[\sigma_x\left(x-S_{Mx}^{*}\right)+\sigma_y\left(y-S_{My}^{*}\right)\right]\left(\sigma_x S_{Mx}^{*}+\sigma_y S_{My}^{*}\right)\le 0$. If a side panel does not satisfy this condition, it is not included in the summation that forms the third term in eqn. (3.13). All the panel integrals that meet this criterion can be evaluated very simply using eqn. (3.10). Taking the five-sided wave model in Figs. 6c or 6d as examples, we see that only two side panels in each of those figures meet this criterion. Taking the three-sided wave model in Fig. 7b as an example, the single side panel that meets this criterion is given by ΔOP_1P_3 .

4) It is self-evident that the contact discontinuity will intersect two side panels in our wave model. Let us label those side panels "$istart\_c$" and "$iend\_c$". Figs. 6c and 6d show examples. We always insist on a counterclockwise direction that connects panel "$istart\_c$" to panel "$iend\_c$". This uniquely identifies panels "$istart\_c$" and "$iend\_c$". The fourth term in eqn. (3.13) integrates over sub-portion of the panel that starts at C_{istart_c} and ends at $P_{cyc(istart\_c+1)}$. Here the point C_{istart_c} identifies the intersection of the contact discontinuity with the panel and the point $P_{cyc(istart\_c+1)}$ identifies the endpoint of the same panel. Eqn. (3.11) shows how this panel integral can be evaluated automatically on a computer. If we take Fig. 7b as an example, this panel integration extends over ΔOP_3Q_4 .

5) The previous point has explained how the panel "$iend\_c$" is identified. The fifth term in eqn. (3.13) is an integral over sub-portion of that panel. Figs. 6c and 6d show examples. Now the point P_{iend_c} is the starting point of that panel and the point C_{iend_c} is the point where the contact discontinuity intersects that panel. Eqn. (3.12) shows how this panel integral can be evaluated automatically on a computer. If we take Fig. 7b as an example, this panel integration extends over ΔOQ_3P_1 .

This completes our intuitive description of eqn. (3.13), which yields the very useful eqn. (3.14).

It is worthwhile observing that eqn. (3.14) of the present paper finds its analogue in eqn. (25) of Balsara [5] for the multidimensional HLLC Riemann solver on structured meshes. Consequently, we have shown that on structured meshes our present multidimensional HLLC Riemann solver reduces exactly to the similar Riemann solver from Balsara [5]. The good properties catalogued in that paper are, therefore, also conferred upon this work.

The multidimensional HLL Riemann solver supplies inputs to its HLLC variant. As a result, it should be implemented before embarking on the Riemann solver described in this Section. The multidimensional HLLC Riemann solver described in this Section is implemented on a computer by following the logic of the Sub-sections, along with their respective equations, in the very same order that they are written here.

III.4) Alternative strategy for Integrating the Conservation Law

Eqn. (3.14), along with eqn. (3.3), give us the state $\mathbf{U}^*_{C1}$ as well as the fluxes $\mathbf{F}^*_{C1}$ and $\mathbf{G}^*_{C1}$ via one possible consistent formulation. This formulation is predicated on extracting density, velocities and pressure from eqns. (2.17) and (3.2) from the HLL state. However, realize that the standard derivation of the one-dimensional HLLC Riemann solver entails equating the pressures from the states $\mathbf{U}^*_{C1}$ and $\mathbf{U}^*_{C2}$. The corresponding velocity across the contact discontinuity is adjusted so as to bring about this equality. The resulting velocities and pressure can be different from the ones in eqns. (2.17) and (3.2). We wish to show that an alternative formulation exists that exactly mimics the one-dimensional formulation.

To that end, please look at Fig. 6 and realize that we can always find l_{UHLLC} , the length of the contact discontinuity that is contained within the wave model. It is easily written as

$$l_{UHLLC} = \sqrt{\left(\mathrm{X}_{istart\_c} - \mathrm{X}_{iend\_c}\right)^2 + \left(\mathrm{Y}_{istart\_c} - \mathrm{Y}_{iend\_c}\right)^2} \tag{3.15}$$

We also define ψ_{UHLLC} , which tells us whether the HLL velocity from eqn. (2.17) is oriented along or opposite to the normal vector for the contact discontinuity. We have

$$\psi_{UHLLC} \equiv \mathrm{sgn}\left(S^*_\sigma\right) \qquad ; \qquad S^*_\sigma \equiv \sigma_x S^*_{Mx} + \sigma_y S^*_{My} \tag{3.16}$$

Eqn. (3.13) can then be written in an equivalent form as

$$\begin{aligned}
&\mathrm{A}_{UHLLC}\mathbf{U}^*_{C1} + \frac{1}{2} l_{UHLLC}\psi_{UHLLC}\left(\sigma_x \mathbf{F}^*_{C1} + \sigma_y \mathbf{G}^*_{C1} - S^*_\sigma \mathbf{U}^*_{C1}\right) = \\
&\quad -\sum_{i=1}^{imax} \frac{1}{T^2}\Big(\text{Area integral over } \mathrm{OP}_i\mathrm{P}_{cyc(i+1)}\Big)_{\text{if}\left[\sigma_x\left(x - S^*_{Mx}\right) + \sigma_y\left(y - S^*_{My}\right)\right]\left(\sigma_x S^*_{Mx} + \sigma_y S^*_{My}\right) \le 0} \\
&\quad -\frac{1}{T^2}\Big(\text{Area integral over } \mathrm{OC}_{istart_c}\mathrm{P}_{cyc(istart_c+1)}\Big) - \frac{1}{T^2}\Big(\text{Area integral over } \mathrm{OP}_{iend_c}\mathrm{C}_{iend_c}\Big)
\end{aligned} \tag{3.17}$$

The above equation can be written very compactly as

$$\mathrm{A}_{UHLLC}\mathbf{U}^{*}_{C1}+\frac{1}{2}l_{UHLLC}\psi_{UHLLC}\left(\mathbf{F}^{*}_{\sigma;C1}-S^{*}_{\sigma}\mathbf{U}^{*}_{C1}\right)=\mathbf{R}_{C1}\quad\text{with}\quad\mathbf{F}^{*}_{\sigma;C1}\equiv\sigma_x\mathbf{F}^{*}_{C1}+\sigma_y\mathbf{G}^{*}_{C1}\tag{3.18}$$

Here $\mathbf{F}^{*}_{\sigma;C1}$ is the flux along the contact discontinuity and $\mathbf{R}_{C1}$ is the sum of all the panel integrals that contribute to the outer surface of our wave model on the side of $\mathbf{U}^{*}_{C1}$, i.e., the right hand side of eqn. (3.17). The corresponding equation for $\mathbf{U}^{*}_{C2}$ is given by

$$\left(\mathrm{A}_{UHLL}-\mathrm{A}_{UHLLC}\right)\mathbf{U}^{*}_{C2}-\frac{1}{2}l_{UHLLC}\psi_{UHLLC}\left(\mathbf{F}^{*}_{\sigma;C2}-S^{*}_{\sigma}\mathbf{U}^{*}_{C2}\right)=\mathbf{R}_{C2}\quad\text{with}\quad\mathbf{F}^{*}_{\sigma;C2}\equiv\sigma_x\mathbf{F}^{*}_{C2}+\sigma_y\mathbf{G}^{*}_{C2}\tag{3.19}$$

Realize that $\mathbf{F}^{*}_{\sigma;C1}$ depends on $\mathbf{U}^{*}_{C1}$; likewise $\mathbf{F}^{*}_{\sigma;C2}$ depends on $\mathbf{U}^{*}_{C2}$. Our task is to solve eqns. (3.18) and (3.19) consistent with matching pressures and normal velocity across the contact discontinuity.

For the Euler equations we can now write $\mathbf{U}^{*}_{C1}=\left(\rho^{*C1},\ \rho^{*C1}\mathrm{v}_x^{*C1},\ \rho^{*C1}\mathrm{v}_y^{*C1},\ \rho^{*C1}\mathrm{v}_z^{*C1},\ \varepsilon^{*C1}\right)^{T}$ with a corresponding expression for the flux $\mathbf{F}^{*}_{\sigma;C1}$. By making the transcription $C1\rightarrow C2$ we can obtain similar expressions for $\mathbf{U}^{*}_{C2}$ and $\mathbf{F}^{*}_{\sigma;C2}$. We can make a further internal rotation between the second and third components of $\mathbf{U}^{*}_{C1}$ to get

$$\mathrm{v}^{*}_{\sigma}=\sigma_x\mathrm{v}_x^{*C1}+\sigma_y\mathrm{v}_y^{*C1}\quad;\quad\mathrm{v}_{\kappa}^{*C1}=\sigma_y\mathrm{v}_x^{*C1}-\sigma_x\mathrm{v}_y^{*C1}\tag{3.20}$$

Notice that v^{*}_{σ} does not have a superscript of "C1" because our eventual goal is to have the same normal velocity on either side of the contact. Similarly, the pressure P^{*} is free of a superscript of "C1". Eqn. (3.18) can now be written as

$$\mathrm{A}_{UHLLC}\begin{pmatrix}\rho^{*C1}\\ \rho^{*C1}\mathrm{v}^{*}_{\sigma}\\ \rho^{*C1}\mathrm{v}_{\kappa}^{*C1}\\ \rho^{*C1}\mathrm{v}_z^{*C1}\\ \varepsilon^{*C1}\end{pmatrix}+\frac{1}{2}l_{UHLLC}\psi_{UHLLC}\begin{pmatrix}\rho^{*C1}\left(\mathrm{v}^{*}_{\sigma}-S^{*}_{\sigma}\right)\\ \rho^{*C1}\mathrm{v}^{*}_{\sigma}\left(\mathrm{v}^{*}_{\sigma}-S^{*}_{\sigma}\right)+\mathrm{P}^{*}\\ \rho^{*C1}\mathrm{v}_{\kappa}^{*C1}\left(\mathrm{v}^{*}_{\sigma}-S^{*}_{\sigma}\right)\\ \rho^{*C1}\mathrm{v}_z^{*C1}\left(\mathrm{v}^{*}_{\sigma}-S^{*}_{\sigma}\right)\\ \varepsilon^{*C1}\left(\mathrm{v}^{*}_{\sigma}-S^{*}_{\sigma}\right)+\mathrm{P}^{*}\mathrm{v}^{*}_{\sigma}\end{pmatrix}=\begin{pmatrix}r_1^{C1}\\ r_{\sigma}^{C1}\\ r_{\kappa}^{C1}\\ r_4^{C1}\\ r_5^{C1}\end{pmatrix}\tag{3.21}$$

Eqn. (3.19) can now be written as

$$\left(\mathrm{A}_{UHLL}-\mathrm{A}_{UHLLC}\right)\begin{pmatrix}\rho^{*C2}\\ \rho^{*C2}\mathrm{v}_{\sigma}^{*}\\ \rho^{*C2}\mathrm{v}_{\kappa}^{*C2}\\ \rho^{*C2}\mathrm{v}_{z}^{*C2}\\ \varepsilon^{*C2}\end{pmatrix}-\frac{1}{2}l_{UHLLC}\psi_{UHLLC}\begin{pmatrix}\rho^{*C2}\left(\mathrm{v}_{\sigma}^{*}-S_{\sigma}^{*}\right)\\ \rho^{*C2}\mathrm{v}_{\sigma}^{*}\left(\mathrm{v}_{\sigma}^{*}-S_{\sigma}^{*}\right)+\mathrm{P}^{*}\\ \rho^{*C2}\mathrm{v}_{\kappa}^{*C2}\left(\mathrm{v}_{\sigma}^{*}-S_{\sigma}^{*}\right)\\ \rho^{*C2}\mathrm{v}_{z}^{*C2}\left(\mathrm{v}_{\sigma}^{*}-S_{\sigma}^{*}\right)\\ \varepsilon^{*C2}\left(\mathrm{v}_{\sigma}^{*}-S_{\sigma}^{*}\right)+\mathrm{P}^{*}\mathrm{v}_{\sigma}^{*}\end{pmatrix}=\begin{pmatrix}r_{1}^{C2}\\ r_{\sigma}^{C2}\\ r_{\kappa}^{C2}\\ r_{4}^{C2}\\ r_{5}^{C2}\end{pmatrix} \tag{3.22}$$

By making suitable combinations of the first and second rows of eqns. (3.22) we get the following pair of linear equations

$$\frac{1}{2}l_{UHLLC}\psi_{UHLLC}\mathrm{P}^{*}+r_{1}^{C1}\mathrm{v}_{\sigma}^{*}=r_{\sigma}^{C1}\quad;\quad-\frac{1}{2}l_{UHLLC}\psi_{UHLLC}\mathrm{P}^{*}+r_{1}^{C2}\mathrm{v}_{\sigma}^{*}=r_{\sigma}^{C2} \tag{3.23}$$

The above two equations can be solved for P^{*} and v_{σ}^{*} . Since r_{1}^{C1} and r_{1}^{C2} are mass fluxes, they can be guaranteed to have the same sign over surfaces that are expanding with an extremal speed. This, in turn, guarantees that eqn. (3.23) will always have a solution. The first rows of eqns. (3.21) and (3.22) can be used to obtain the densities ρ^{*C1} and ρ^{*C2} . The third and fourth rows of eqns. (3.21) and (3.22) can be used to obtain the transverse velocities $\mathrm{v}_{\kappa}^{*C1}$, $\mathrm{v}_{\kappa}^{*C2}$, v_{z}^{*C1} and v_{z}^{*C2} . The fifth rows of eqns. (3.21) and (3.22) can be used to obtain the total energies ε^{*C1} and ε^{*C2} . The rotation in eqn. (3.20) can be undone and the states $\mathbf{U}_{C1}^{*}$ and $\mathbf{U}_{C2}^{*}$ as well as the fluxes $\mathbf{F}_{C1}^{*}$, $\mathbf{G}_{C1}^{*}$, $\mathbf{F}_{C2}^{*}$ and $\mathbf{G}_{C2}^{*}$ can be assembled. This completes our discussion of the HLLC Riemann solver for Euler flow. The analogous equations for MHD are given in Appendix D.

A further benefit of the formulation in this Sub-section arises from realizing that we get both the states and their associated fluxes. As a result, if the present Riemann solver is used in an Arbitrary Lagrangian Eulerian (ALE) formulation, we can always identify the flux along a moving edge. It is simply a matter of evaluating which side of the contact discontinuity the vertex of the ALE mesh lies. Such evaluations have been described in great detail in Sub-section III.1.

Also notice that we started this Sub-section with a conjectured contact velocity $S_{x}^{*}\mathbf{x}+S_{y}^{*}\mathbf{y}$ which was obtained from the two-dimensional HLL Riemann solver. After application of the steps described here, we have an improved contact velocity $\mathrm{v}_{\sigma}^{*}\left(\sigma_{x}\mathbf{x}+\sigma_{y}\mathbf{y}\right)$. In principle, the panel integrals can be evaluated again and the procedure described here can be repeated to convergence. In practice, that has been found to be unnecessary. We also note that when convergence is obtained we get a consistency between the two HLLC states and the HLL state from eqn. (2.8). The consistency condition can be explicitly written as

$$\mathrm{A}_{UHLLC}\mathbf{U}_{C1}^{*}+\left(\mathrm{A}_{UHLL}-\mathrm{A}_{UHLLC}\right)\mathbf{U}_{C2}^{*}=\mathrm{A}_{UHLL}\mathbf{U}^{*} \tag{3.24}$$

A similar consistency exists for the one-dimensional HLLC Riemann solver and the above equation extends it to multiple space dimensions.

IV) Implementation-Related Details

We now describe the logical steps for implementing the multidimensional HLLC Riemann solver. The first five steps describe the implementation of the multidimensional HLL Riemann solver; the next steps describe how the multidimensional HLLC Riemann solver is built on top of it. The description is restricted to a non-moving mesh. The steps are as follows:

1) Following Section II.1 and Appendix A, obtain the intermediate states for the one-dimensional HLL and HLLC Riemann solvers. Obtain all intermediate states and fluxes.

2) Following Sub-section II.2 and Appendix B, obtain the wave model. This is tantamount to obtaining all the wave speeds S_i in the principal directions $\boldsymbol{\eta}_i$. Using eqns. (2.3) and (2.4) we can find the boundary of the strongly interacting state $\mathbf{U}^*$. If the strongly interacting state circumscribes the time axis, we say that we have a subsonic case and follow steps 3) and 4) below. If not, we say that we have a supersonic case and follow step 5) below. It is also worth pointing out that Appendix B describes an all-subsonic variant of the Riemann solver.

3) Since the strongly interacting state is subsonic, follow the steps in Sub-section II.3.a to obtain the strongly interacting state $\mathbf{U}^*$.

4) Since the strongly interacting state is subsonic, follow the steps in Sub-sections II.3.b and II.3.c to obtain the multidimensional HLL fluxes $\mathbf{F}^*$ and $\mathbf{G}^*$ that correspond to the multidimensional HLL state $\mathbf{U}^*$.

5) If the strongly interacting state is supersonic, use the steps in Sub-section II.3.d to find the corresponding multidimensional HLL fluxes. This completes the description of the multidimensional HLL Riemann solver. A step analogous to Sub-section II.3.d also gives us the HLLC fluxes in the supersonic limit.

6) Using eqns. (2.17) and (3.2), find the density ρ^*, the velocities S_{Mx}^* and S_{My}^*, and the pressure P^*, for the strongly interacting state. This allows us to write the propagating wave for the contact discontinuity from eqn. (3.1). Using eqn. (3.3) we can also write the HLLC fluxes $\mathbf{F}_{C1}^*$ and $\mathbf{G}_{C1}^*$, in terms of the HLLC state $\mathbf{U}_{C1}^*$. This linearization is very useful in the derivation of the multidimensional HLLC Riemann solver.

7) Using eqn. (3.4) and the remaining conditionals in Sub-section III.1, find "$i = istart\_c$" and "$i = iend\_c$" in order to find the state $\mathbf{U}_{C1}^*$ that straddles the time axis.

8) Use eqns. (3.8) and (3.9) from Sub-section III.2 to find the area A_{UHLLC} and the area vector $N_x^{C1C2}\mathbf{x} + N_y^{C1C2}\mathbf{y} + N_t^{C1C2}\mathbf{t}$.

9a) If the variant from Sub-section III.3 is used, eqn. (3.14) can now be used to find the state $\mathbf{U}^*_{C1}$. Eqn. (3.3) can then be used to obtain the fluxes $\mathbf{F}^*_{C1}$ and $\mathbf{G}^*_{C1}$. This completes the description of the multidimensional HLLC Riemann solver.

9b) If the variant from Sub-section III.4 is used, eqns. (3.15) and (3.16) are used first and their outputs are then used in eqn. (3.23) to obtain P^* and v^*_σ . With these in hand, the additional rows of eqns. (3.21) and (3.22) can be used to obtain the remaining variables in the states $\mathbf{U}^*_{C1}$ and $\mathbf{U}^*_{C2}$. The corresponding fluxes are easily computed. This variant is better suited for ALE computations. We have also found it to be slightly better performer and have used it in a majority of the results obtained in the next section. This completes the description of the multidimensional HLLC Riemann solver.

It is also worth documenting that the multidimensional HLLC Riemann solver is only invoked if we know that there is a contact discontinuity in the solution. Thus, we evaluate the undivided difference of the density and require it to exceed some small fraction of the density (say 5%). It is also useful to ensure that the density is a genuine contact discontinuity and not a shock. For that reason, we evaluate a flattener variable, as described in Balsara [13], and only invoke the HLLC Riemann solver if no strong shocks are detected. If strong shocks are detected, we locally switch to the multidimensional HLL Riemann solver.

V) Results for Euler Flow

We present a large number of challenging problems for Euler flow in this section. All simulations were run on unstructured meshes. We have used ADER-WENO schemes on the primal mesh. Since the Riemann solver fully accounts for any angular resolution that is available in the underlying mesh, the results in this section also highlight the versatility of the multidimensional Riemann solver design presented here.

Within our higher order ADER-WENO schemes we have used a three-point Simpson quadrature formula in space for integrating the numerical fluxes across a zone boundary, while Gauss-Legendre quadrature of appropriate order was employed for computing the integrals in time. In this framework, the multidimensional Riemann solvers are applied at the vertices of the mesh (the two end points of the Simpson formula), while classical one-dimensional Riemann solvers are used at the midpoints of the zone boundaries (the midpoint of the Simpson formula). We emphasize that the zone vertices are the very locations where problems that are simulated with higher order schemes could provoke a loss of density or pressure positivity. For that reason, we strongly recommend that positivity preserving reconstruction strategies be used along with these Riemann solvers, see Balsara [13]. For each test problem we quote the CFL number that it was run with. For our purposes, the timestep restriction from each element is given by the CFL number times the ratio of the in-circle diameter of that element divided by the maximal signal speed in that element.

V.1) Isentropic Vortex – Accuracy Analysis

The isentropic vortex problem, presented by Jiang & Shu [38], consists of a fluid vortex propagating at 45° in a domain with periodic boundaries given by $[-5,5]\times[-5,5]$. We do not describe the details of the problem set-up here since they are very well-known by now. The vortex has continuous and differentiable flow variables making it an ideal candidate for accuracy analysis. We have simulated this problem on unstructured meshes with ADER-WENO schemes (Dumbser & Käser [26], Dumbser *et al*. [27], Balsara *et al*. [12], Balsara *et al*. [14]). The ADER-WENO schemes can access higher orders.

As described above, all the schemes used the multidimensional HLLC Riemann solver described here at the vertices of the mesh along with edge-centered one-dimensional HLLC Riemann solver with the weights assigned via the Simpson rule. All the simulations were run with a CFL of 0.95. The original ADER-WENO scheme had a CFL limit of 0.5 in two-dimensions, so the use of the present Riemann solver technology has indeed doubled the CFL limit for this class of schemes. This expanded CFL for the ADER-WENO algorithm is made possible by the multidimensional fluxes that the present Riemann solver endows to the ADER-WENO scheme. If we do not use a multidimensional Riemann solver (i.e. if we rely exclusively on usual edge-centered one-dimensional Riemann solvers), the ADER-WENO scheme blows up at a CFL number of 0.95.

Tables I a, b and c show the L_1 , L_2 and L_∞ errors in the density variable and the order property for the ADER-WENO schemes at second, third and fourth order. In Tables I *Ne* is a measure of the mesh resolution and denotes the number of elements used per space dimension. In all instances the computation has been run with a CFL of 0.95 and we see that the order property is met with increasing resolution. The use of the multidimensional HLLC Riemann solver only slightly increases the computational complexity per time-step for the ADER-WENO schemes. However, the doubling of the CFL number shortens the overall CPU time by a factor of up to 40%. The L_p error norms have been computed as

$$L_p = \sqrt[p]{\int_\Omega \left(\left| w_h - U_e \right|^p \right) dx\, dy}$$

where w_h represents the reconstructed solution and U_e is the exact one, which for this problem is given by the initial condition after one period of advection. The integral in the above formula is approximated with a Gaussian quadrature formula of suitable order of accuracy.

In Tables Id, Ie and If we give a detailed quantitative comparison of the L_1 error norms obtained with a classical one-dimensional HLL Riemann solver at a CFL of 0.475 with the new 2D HLL Riemann solver at a CFL of 0.475 and 0.95, respectively. From Tables Id to If we see that the errors are comparable. Tables Id, Ie and If also show the CPU run times of the different runs at each resolution. We see that the multidimensional Riemann solver running with a CFL of 0.95 is a good 35% faster than the one-dimensional Riemann solver running with a CFL of 0.475. While CPU times depend on the overall code structure and memory usage, it is worth pointing out that there are potentially much larger performance gains to be had by solving the multidimensional Riemann problem on a GPU. We leave that exploration for later. We do,

however, get the overarching insight that the added cost of the multidimensional Riemann solver is offset by the larger CFL number that it enables.

TABLE Ia – Errors in the density for the isentropic vortex using ADER-WENO scheme at second order with 2d HLLC Riemann Solver

#of elements, 1d	L_1 Error	L_1 Order	L_2 Error	L_2 Order	L_∞ Error	L_∞ Order
64	2.2707E-01		3.7415E-02		2.4855E-02	
128	5.1411E-02	2.14	8.1563E-03	2.20	6.7079E-03	1.89
256	1.3657E-02	1.91	2.1622E-03	1.92	1.8540E-03	1.86
512	3.5597E-03	1.94	5.7674E-04	1.91	4.4355E-04	2.06

TABLE Ib – Errors in the density for the isentropic vortex using ADER-WENO scheme at third order with 2d HLLC Riemann Solver

#of elements, 1d	L_1 Error	L_1 Order	L_2 Error	L_2 Order	L_∞ Error	L_∞ Order
64	7.0733E-02		1.7641E-02		1.0994E-02	
128	9.9983E-03	2.82	2.5492E-03	2.79	1.5428E-03	2.83
256	1.2705E-03	2.98	3.2764E-04	2.96	1.9774E-04	2.96
512	1.5977E-04	2.99	4.1369E-05	2.99	2.5654E-05	2.95

TABLE Ic – Errors in the density for the isentropic vortex using ADER-WENO scheme at fourth order with 2d HLLC Riemann Solver

#of elements, 1d	L_1 Error	L_1 Order	L_2 Error	L_2 Order	L_∞ Error	L_∞ Order
64	9.1699E-03		1.8107E-03		1.2780E-03	
128	4.6866E-04	4.29	9.3267E-05	4.28	8.7567E-05	3.87
256	2.8738E-05	4.03	5.7309E-06	4.02	5.8208E-06	3.91
512	1.7730E-06	4.02	3.5168E-07	4.03	3.5236E-07	4.05

TABLE Id – Errors in the density for the isentropic vortex using ADER-WENO scheme at second order with 1d HLL Riemann Solver at CFL=0.475, the 2d HLLC Riemann Solver at CFL=0.475 and the 2d HLLC Riemann Solver at CFL=0.95.

	1D HLL, CFL=0.475			2d HLL, CFL=0.475			2d HLL, CFL=0.95		
Ne	L_1 Error	$O(L_1)$	t_{CPU} [s]	L_1 Error	$O(L_1)$	t_{CPU} [s]	L_1 Error	$O(L_1)$	t_{CPU} [s]
32	5.2566E-01		0.88	7.2058E-01		3.30	7.0295E-01		0.64
64	1.7797E-01	1.56	6.37	2.7255E-01	1.40	8.58	2.7082E-01	1.38	4.30
128	4.4227E-02	2.01	45.60	7.7186E-02	1.82	60.82	7.5899E-02	1.84	29.96
256	1.0199E-02	2.12	354.56	1.7585E-02	2.13	462.13	1.7382E-02	2.13	226.55

TABLE Ie – Errors in the density for the isentropic vortex using ADER-WENO scheme at third order with 1d HLL Riemann Solver at CFL=0.475, the 2d HLLC Riemann Solver at CFL=0.475 and the 2d HLLC Riemann Solver at CFL=0.95.

	1D HLL, CFL=0.475			2d HLL, CFL=0.475			2d HLL, CFL=0.95		
Ne	L_1 Error	$O(L_1)$	t_{CPU} [s]	L_1 Error	$O(L_1)$	t_{CPU} [s]	L_1 Error	$O(L_1)$	t_{CPU} [s]
32	2.4313E-01		2.27	3.8393E-01		2.79	3.6314E-01		1.55
64	4.7679E-02	2.35	16.36	7.6082E-02	2.34	20.80	7.2603E-02	2.32	11.16
128	6.5599E-03	2.86	116.69	1.0904E-02	2.80	149.20	1.0334E-02	2.81	73.80
256	8.1866E-04	3.00	836.24	1.3665E-03	3.00	1059.97	1.2933E-03	3.00	551.36

TABLE If – Errors in the density for the isentropic vortex using ADER-WENO scheme at fourth order with 1d HLL Riemann Solver at CFL=0.475, the 2d HLLC Riemann Solver at CFL=0.475 and the 2d HLLC Riemann Solver at CFL=0.95.

	1D HLL, CFL=0.475			2d HLL, CFL=0.475			2d HLL, CFL=0.95		
Ne	L_1 Error	$O(L_1)$	t_{CPU} [s]	L_1 Error	$O(L_1)$	t_{CPU} [s]	L_1 Error	$O(L_1)$	t_{CPU} [s]
32	7.1406E-02		4.03	1.1578E-01		5.11	1.1287E-01		2.85
64	6.0145E-03	3.57	31.88	1.0148E-02	3.51	39.97	9.9035E-03	3.51	20.39
128	3.9916E-04	3.91	203.62	6.4695E-04	3.97	259.63	6.3566E-04	3.96	142.26
256	2.0140E-05	4.31	1488.95	3.3743E-05	4.26	1907.21	3.3459E-05	4.25	958.74

V.2) Sod and Lax Problems on a Two-dimensional Mesh

The well-known Sod and Lax problems were run on a two-dimensional mesh using the third order ADER-WENO scheme. This test was designed to demonstrate the fact that the multi-dimensional HLLC Riemann solver described here indeed allows a larger CFL number. To that end, we ran all the problems here with a CFL number of 0.95.

Fig. 8a shows the density along with the mesh for the Sod problem when the facially averaged numerical fluxes were obtained by combining the vertex-centered multidimensional HLLC Riemann solver with the edge-centered one-dimensional HLLC Riemann solver. The weights were assigned via the Simpson rule so that the resulting flux is up to fourth order accurate. Fig. 8a shows that the simulation ran stably. Fig. 8b shows the result of running the same simulation on the same mesh with the same CFL. The only difference between Figs. 8a and 8b is that the latter simulation was run by relying exclusively on the one-dimensional HLLC Riemann solver. We see that the simulation has gone unstable at a CFL of 0.95 when the numerical fluxes rely only on the one-dimensional Riemann solver. At a CFL of 0.5, the simulation in Fig. 8b runs stably. This shows that the inclusion of the multidimensional HLLC Riemann solver has contributed to the stability of the scheme. Fig. 8c shows that the density from Fig. 8a plotted out in one-dimension. In Fig. 8c we also show the density from two other simulations – with multidimensional Riemann solver running with CFL 0.475 and with one-dimensional Riemann solver running with CFL 0.475. We see that all simulations perform equally well. The exact solution is shown with a solid line while the symbols show the numerical solution. We see that the contact discontinuity is well-resolved.

To present the reader with a fair timing comparison, we ran the Sod shock problem in three different ways using a third order ADER-WENO scheme. We ran it with the multidimensional HLLC Riemann solver and a CFL of 0.95 to obtain a run time of 10.08 CPU seconds. We ran it with the multidimensional HLLC Riemann solver and a CFL of 0.475 to obtain a run time of 19.94 CPU seconds. We ran it with the one-dimensional HLLC Riemann solver and a CFL of 0.475 to obtain a run time of 13.54 CPU seconds. We see that the multidimensional Riemann solver running with a larger CFL number indeed gives us a 35% advantage over a one-dimensional Riemann solver. The added cost of the multidimensional Riemann solver is offset by the larger CFL number that it enables.

The simulation shown in Fig. 9a was run with conditions analogous to those in Fig. 8a and shows the density along with the mesh for the Lax problem. When the same simulation was run on the same mesh with a CFL of 0.95 but with the one-dimensional HLLC Riemann solver contributing exclusively to the numerical flux, the code crashed. We did verify that the same code was stable and salient for the Lax problem at a CFL of 0.45. This again underscores the extra time-step stability provided by the vertex-centered multidimensional HLLC Riemann solver. Fig. 9b is a one-dimensional plot of the density from Fig. 9a and shows that the contact discontinuity is well-resolved. The reference solution is shown with a solid line while the legends show the computed solution.

V.3) Two-Dimensional Riemann Problems

In recent years, the two-dimensional Riemann problems suggested by Schulz-Rinne *et al.* [52] have become increasingly important as test problems. We show two of those test problems here. Since both these problems were described in detail in the recent paper by Balsara [5], we do not repeat the description of the problem set-up over here. The problems were run with the multi-dimensional HLL and HLLC Riemann solvers using the third order ADER-WENO scheme at a CFL of 0.95. An h=1/1000 unstructured mesh with ~2.3 million elements was used. The stopping times for the first and second Riemann problems were 0.5 and 1.05, respectively.

Fig. 10a shows the density from the first two-dimensional Riemann problem run with the third-order ADER-WENO scheme along with the multidimensional HLLC Riemann solver. As a counterfoil, Fig. 10b shows an entirely similar simulation but with the multidimensional HLL Riemann solver. By comparing Figs. 10a and 10b we see that the use of the HLLC Riemann solver has made it possible for us to capture a much more pronounced Kelvin-Helmholtz instability roll-up in Fig. 10a. This highlights the better resolution afforded by the HLLC algorithm over the HLL algorithm, especially in multiple dimensions. The multidimensional HLLC Riemann solver can adjust to the locally changing strength and direction of the contact discontinuity, thereby keeping the contact discontinuity extremely sharp. When the third order ADER-WENO code was used with the multidimensional HLLC Riemann solver on an h=1/400 unstructured mesh, we were still able to capture the vortex sheet roll-up, showing that the resolving power of the Riemann solver enables us to capture features of interest even on resolution-starved meshes.

Fig. 11a shows the density from the second two-dimensional Riemann problem run with the third-order ADER-WENO code along with the multidimensional HLLC Riemann solver. For

comparison, Fig. 11b shows an entirely similar simulation but with the multidimensional HLL Riemann solver. For this problem it is traditional to show the lower left corner of the mesh because it illustrates the roll-up of the very prominent Mach stem that forms. Consequently, Fig. 11 shows such a portion of the mesh as described in Balsara [5]. We see from Fig. 11a that the Mach stem has shown a very beautiful roll-up, whereas the roll-up is not as prominent in Fig. 11b. As with the first two-dimensional Riemann problem, an h=1/400 unstructured mesh would still capture the Mach stem roll-up when a third order scheme is used in conjunction with the Riemann solver technology presented here.

V.4) Double Mach Reflection Problem

This problem, which was described in Woodward and Colella [61], consists of a Mach 10 shock wave impinging on a wedge. The wedge makes an angle of 30^o with the horizontal axis. The interaction of the shock and the wedge sets up the double Mach structure. The problem was set up using an unstructured mesh with h=1/400 and ~1.4 million elements. The problem was run to a final time of 0.2 using the scheme described here and all simulations used a CFL of 0.95. Because the problem was done on an unstructured mesh that can conform to arbitrary geometries, we used a rightward-propagating Mach 10 strong shock that actually impinges on an inclined wedge. Since this problem is very well described in the literature, we do not repeat the description here.

Fig. 12a shows the density from the double Mach reflection problem run with the third-order ADER-WENO code along with the multidimensional HLLC Riemann solver. Fig. 12b zooms in on the roll-up of the Mach stem. Cockburn & Shu [21] showed that they needed a fourth order scheme operating at 1920×480 zone resolution on a structured mesh to capture the Kelvin-Helmholtz instability of the Mach stem. Despite being third order, our h=1/400 ADER-WENO simulation with the multidimensional HLLC Riemann solver in Fig. 12 has already picked up the Kelvin-Helmholtz instability at the Mach stem. We see, therefore, that the inclusion of true multidimensionality has enabled the third order schemes to become quite competitive with their fourth order discontinuous Galerkin cousins. It should also be noted that all our simulations were run with a CFL of 0.95 while the fourth order RKDG scheme of Cockburn & Shu [21] can only sustain a maximal CFL of 0.1. The use of Lax-Wendroff procedures in the predictor step without multidimensional Riemann solvers will not change this adverse timestep scaling for DG schemes. This also makes a compelling case for including multidimensional Riemann solvers in Godunov schemes that go beyond second order, a promise that is fulfilled for the first time in this paper.

Figs. 12c and 12d show analogous information to Figs. 12a and 12b. However, Figs. 12c and 12d show the results from a third-order ADER-WENO code along with a one-dimensional HLLC Riemann solver run with CFL 0.475. Fig. 12 shows that the results in Figs. 12a and 12b are visually analogous to the results in Figs. 12c and 12d. We have also run this problem with the multidimensional HLLC Riemann solver with a CFL of 0.475 and found the results to look the same. The simulation with multidimensional Riemann solver and CFL 0.95 took 19139 CPU seconds to run. The simulation with multidimensional Riemann solver and CFL 0.475 took 37645 CPU seconds to run. The simulation with the one dimensional Riemann solver and CFL 0.475 took 31991CPU seconds to run; and would have gone unstable if run with a higher CFL.

We therefore see all over again that the multidimensional Riemann solver running with a larger CFL number indeed gives us a 35% advantage over a one-dimensional Riemann solver.

V.5) Forward Facing Step Problem

This problem, which was also described in Woodward and Colella [61], consists of a Mach 3 flow impinging on a forward facing step. A bow shock emanates from the step and interacts with the upper wall in the problem. This interaction sets up a normal shock and a slip surface. The interaction of the curved bow shock, the normal shock and the slip surface sets up a triple point structure in the flow. The problem was set up using an unstructured mesh with h=1/400 and ~0.9 million elements. The problem was run to a final time of 2.5 with a CFL of 0.95. Since this problem is very well described in the literature, we do not repeat the description here.

Fig. 13 shows the density from the forward facing step problem run with the third-order ADER-WENO code along with the multidimensional HLLC Riemann solver. The unstructured mesh technology was able to provide high quality angular resolution at the corner of the forward-facing step. As a result, we did not need to use the enthalpy fix-ups described in Woodward and Colella [61]. The Riemann solver naturally responds to the higher resolution of the mesh, thus providing a more physical and high-quality solution at the forward-facing step. This ability of the present multidimensional Riemann solver to respond to the quality of the mesh is one of the strongest features of the work presented here.

V.6) Blast Problem in Two Dimensions

We solve a 2D blast wave problem on a circular computational domain with unit radius. The initial condition is given by

$$U(x,0) = \begin{cases} U_i & \text{if } r \le 0.5 \\ U_0 & \text{if } r > 0.5 \end{cases}$$

with $r^2 = x^2 + z^2$, hence a circle of radius r=0.5 separates the inner state U_i from the outer state U_o. For the inner state we take $\rho_i=1.0$, $u_i=v_i=0$, $p_i=1$ and for the outer state we use $\rho_o=0.125$, $u_o=v_o=0$, $p_o=0.1$. The ratio of specific heats is taken to be $\gamma = 1.4$. The problem is solved on a two-dimensional triangular mesh composed of 68,324 triangles with characteristic mesh spacing h=1/100. The CFL number is set to unity and a third order ADER-WENO scheme is used together with the multi-dimensional HLLC Riemann solver. A reliable 1D reference solution can be obtained for this problem by solving an equivalent 1D PDE system with geometric reaction source term using the rotational symmetry of the problem. It has been computed using a second order TVD scheme with the Osher-type flux of Dumber and Toro [28] on 5000 radial grid points.

The numerical results obtained with the third order unstructured ADER-WENO scheme employing the multi-dimensional HLLC Riemann solver are depicted in Fig. 14 and show a very good agreement with the 1D reference solution. Fig. 14a shows the density as a vertical height

with the two-dimensional mesh overlaid. Fig. 14b shows the angularly averaged density as a function of radius. Figs. 14c and 14d show the same for the pressure and velocity.

VI) Results for MHD Flow

We present a large number of challenging problems for MHD flow in this section. All simulations were run on unstructured meshes. We have used ADER-WENO schemes on primal meshes. Since the unstructured mesh code base did not include support for divergence-free reconstruction of magnetic fields (Balsara [9], [10], [11]), we used the formulation by Dedner *et al*. [25] instead.

VI.1) Isentropic MHD Vortex – Accuracy Analysis

The isentropic MHD vortex problem, presented by Balsara [10], consists of a magnetofluid vortex propagating at 45° in a domain with periodic boundaries given by $[-5,5]\times[-5,5]$. We do not describe the details of the problem set-up here since they are very well-documented in the original reference. The vortex has continuous and differentiable fluid and magnetic field variables making it an ideal candidate for accuracy analysis. We have run this problem with the ADER-WENO scheme with a CFL of 0.95. The generalized Lagrange multiplier field was made to propagate with a speed of 2.0.

Tables II a, b and c show the L_1 , L_2 and L_∞ errors in the x-component of the magnetic field. The tables document the order property for the ADER-WENO-based MHD schemes at second, third and fourth order. In all instances we see that the order property is met with increasing resolution.

TABLE IIa – Errors in the x-magnetic field for the MHD vortex using ADER-WENO scheme at second order with 2d HLLC Riemann Solver

#of elements, 1d	L_1 Error	L_1 Order	L_2 Error	L_2 Order	L_∞ Error	L_∞ Order
32	1.5190E+00		3.5249E-01		2.2613E-01	
64	3.2916E-01	2.21	8.5632E-02	2.04	5.8062E-02	1.96
128	6.3025E-02	2.38	1.6462E-02	2.38	1.3349E-02	2.12
256	1.3730E-02	2.20	3.8460E-03	2.10	4.1990E-03	1.67

TABLE IIb – Errors in the x-magnetic field for the MHD vortex using ADER-WENO scheme at third order with 2d HLLC Riemann Solver

#of elements, 1d	L_1 Error	L_1 Order	L_2 Error	L_2 Order	L_∞ Error	L_∞ Order
32	9.5889E-01		2.3595E-01		1.5097E-01	
64	1.5092E-01	2.67	4.1727E-02	2.50	2.9351E-02	2.36
128	1.9662E-02	2.94	5.6241E-03	2.89	4.0946E-03	2.84

256	2.4170E-03	3.02	6.9451E-04	3.02	5.1535E-04	2.99

TABLE IIc – Errors in the x-magnetic field for the MHD vortex using ADER-WENO scheme at fourth order with 2d HLLC Riemann Solver

#of elements, 1d	L_1 Error	L_1 Order	L_2 Error	L_2 Order	L_∞ Error	L_∞ Order
32	1.5822E-01		4.1676E-02		3.0719E-02	
64	7.3432E-03	4.43	2.0130E-03	4.37	1.5054E-03	4.35
128	4.5637E-04	4.01	1.2777E-04	3.98	1.0978E-04	3.78
256	3.5203E-05	3.70	1.0015E-05	3.67	1.6566E-05	2.73

VI.2) Magnetized Rotor Problem

This well-known MHD problem was first documented in Balsara & Spicer [7] and also Balsara [10]. It consists of a central, uniformly rotating vortex in a non-rotating ambient medium. An initially uniform magnetic field threads through both regions. The details are described in the above-mentioned references. A circular region with radius 0.5 was triangulated using 129,596 elements. Since the dense, rotating vortex extends out to a radius of 0.1, we used h=0.0025 for a radius out to 0.13 and h=0.005 for larger radii. A third order ADER-WENO scheme was used with a CFL of 0.95. The generalized Lagrange multiplier field was made to propagate with a speed of 2.0. Fig. 15 shows the final result for the rotor problem, at a time of 0.25. Figs. 15a, 15b, 15c and 15d show the density, pressure, magnetic pressure and Mach number at the final time. All the requisite MHD flow features are nicely captured in our simulations.

VI.3) Orszag-Tang Problem

This well-known problem by Orszag & Tang [44] was initialized on a periodic domain spanning $[0,2]\times[0,2]$. It was run to a stopping time of unity with a third order accurate ADER-WENO scheme using a CFL of 0.95. The generalized Lagrange multiplier field was made to propagate with a speed of 2.5. Figs. 16a, 16b, 16c and 16d show the final density, pressure, magnitude of the velocity and the magnitude of the magnetic field respectively for the Orszag-Tang problem. We see that the density and pressure have remained positive. The simulation forms a current sheet with oppositely oriented x-components of magnetic field in the center of the computational domain, as can be surmised from Fig. 16d. The velocity field also shows fluid squirting out in the positive and negative x-directions at the location of the current sheet.

VII) Conclusions

In this paper we have presented a two-dimensional HLLC Riemann solver for unstructured meshes. It is the analogue of the two-dimensional HLLC Riemann solver for

structured meshes that was presented in Balsara [5]. It is applied at the vertices of a mesh. It accepts as input all the states that come together at that vertex as well as the angular structure of the mesh around that vertex. As output, it provides the resolved state as well as the two fluxes in the x- and y-directions. In addition, we enunciate four general principles for the design of such Riemann solvers: 1) The multidimensional wave model should be self-similar in space-time. 2) It should be consistent with the underlying conservation law. 3) It should be entropy enforcing. 4) Physical sub-structures, like the contact discontinuity, should be able to propagate at an arbitrary direction to the mesh while retaining consistency with the underlying conservation law.

By incorporating these design principles on an unstructured mesh, we obtain a Riemann solver that accounts for the angular structure of the mesh as well as the orientation of the contact discontinuity on that mesh. Closed-form expressions are provided for all the expressions, facilitating implementation in numerical codes. The formulation accommodates any number of input states coming together at a vertex. It also accounts for the mesh geometry around that vertex.

The versatility of the method is demonstrated by integrating this Riemann solver in ADER-WENO codes. The ADER-WENO algorithm extends to all orders and is implemented on the primal mesh of a triangulated domain. Several stringent applications drawn from Euler and MHD flow are presented, showing that the method works very well on unstructured meshes. The present multidimensional Riemann solver is cost-competitive with traditional, one-dimensional Riemann solvers. It offers the twin advantages of isotropic propagation of flow features and a larger CFL number.

Acknowledgements

DSB acknowledges support via NSF grants NSF-AST-1009091 and NSF-ACI-1307369. DSB also acknowledges support via NASA grants from the Fermi program as well as NASA-NNX 12A088G. The majority of simulations were performed on a cluster at UND that is run by the Center for Research Computing. Computer support on NSF's XSEDE computing resources is also acknowledged.

MD has been financed by the European Research Council (ERC) under the European Union's Seventh Framework Programme (FP7/2007-2013) with the research project STiMulUs, ERC Grant agreement no. 278267. MD also acknowledges PRACE for awarding access to the SuperMUC supercomputer based in Munich, Germany at the Leibniz Rechenzentrum (LRZ).

RA has been financed by the European Research Council (ERC) under the European Union's Seventh Framework Programme (FP7/2007-2013) with the research project ADDECCO, ERC Advanced Grant agreement no. 226316.

References

[1] R.Abgrall, *Approximation du problème de Riemann vraiment multidimensionnel des équations d'Euler par une méthode de type Roe, I: La linéarisation*, C.R. Acad. Sci. Ser. I, 319 (1994) 499

[2] R.Abgrall, *Approximation du problème de Riemann vraiment multidimensionnel des équations d'Euler par une méthode de type Roe, II: Solution du problème de Riemann approché*, C.R. Acad. Sci. Ser. I, 319 (1994) 625

[3] R. Abgrall, *On essentially non-oscillatory schemes on unstructured meshes : Analysis and implementation*, J. Comput. Phys., 114 (1994) 45-58

[4] D.S. Balsara, *Multidimensional HLLE Riemann solver; Application to Euler and Magnetohydrodynamic Flows*, J. Comput. Phys., 229 (2010) 1970-1993

[5] D.S. Balsara, *A two-dimensional HLLC Riemann solver for conservation laws: Application to Euler and magnetohydrodynamic flows*, Journal of Computational Physics 231 (2012) 7476-7503

[6] D.S. Balsara, *Linearized formulation of the Riemann problem for adiabatic and isothermal magnetohydrodynamics*, Astrophysical Journal Supplement 116 (1998) 119

[7] D.S. Balsara and D.S. Spicer, *A staggered mesh algorithm using high order Godunov fluxes to ensure solenoidal magnetic fields in magnetohydrodynamic simulations*, Journal of Computational Physics 149 (1999) 270-292

[8] D.S. Balsara and C.-W. Shu, Monotonicity preserving weighted non-oscillatory schemes with increasingly high order of accuracy, Journal of Computational Physics, 160 (2000) 405-452

[9] D.S. Balsara , *Divergence-free adaptive mesh refinement for magnetohydrodynamics*, Journal of Computational Physics 174 (2001) 614-648

[10] D. S. Balsara, *Second-order-accurate schemes for magnetohydrodynamics with divergence-free reconstruction*, Astrophysical Journal Supplement 151 (2004) 149-184

[11] D.S. Balsara, *Divergence-free reconstruction of magnetic fields and WENO schemes for magnetohydrodynamics*, J. Comput. Phys., 228 (2009) 5040-5056

[12] Balsara, D.S., Rumpf, T., Dumbser, M. & Munz, C.-D., *Efficient, high-accuracy ADER-WENO schemes for hydrodynamics and divergence-free magnetohydrodynamics*, Journal of Computational Physics, 228 (2009) 2480

[13] Balsara, D.S., *Self-Adjusting, Positivity Preserving High Order Schemes for Hydrodynamics and Magnetohydrodynamics*, to appear, Journal of Computational Physics, 231 (2012b) 7504-7517

[14] Balsara, D.S., Dumbser, M., Meyer, C., Du, H. & Xu, Z., *Efficient Implementation of ADER schemes for Euler and Magnetohydrodynamic flow on structured meshes – Comparison with Runge-Kutta methods*, Journal of Computational Physics, 235 (2013) 934-969

[15] P. Batten, N. Clarke, C. Lambert and D.M. Causon, *On the choice of wavespeeds for the HLLC Riemann solver*, SIAM J. Sci. Comput., 18, (1997) 1553-1570

[16] S.J. Billett and E.F. Toro, *On WAF-type schemes for multidimensional hyperbolic conservation laws*, Journal of Computational Physics, 130 (1997) 1-24

[17] M.Brio, A.R. Zakharian and G.M. Webb, *Two-dimensional Riemann solver for Euler equations of gas dynamics*, J. Comput. Phys., 167 (2001) 177-195

[18] P. Cargo & G. Gallice, *Roe matrices for ideal MHD and systematic construction of Roe matrices for systems of conservation laws*, J. Comput. Phys., 136 (1997) 446

[19] A. Chakraborty and E.F. Toro. *Development of an approximate Riemann solver for the steady supersonic Euler equations*. The Aeronautical Journal. Vol. 98, pages 325-339, 1994.

[20] A.J. Chorin, *Random choice solutions of hyperbolic systems*, J. Comput. Phys., 22 (1976) 517

[21] B. Cockburn and C.-W. Shu, The Runge-Kutta discontinuous Galerkin method for Conservation Laws V, Journal of Computational Physics 141 (1998) 199-224

[22] P. Colella, *A direct Eulerian MUSCL scheme for gas dynamics*, SIAM, J. Sci. Statist. Comput., 6 (1985) 104

[23] P. Colella, *Multidimensional Upwind methods for hyperbolic conservation laws*, J. Comput. Phys., 87, (1990) 171

[24] P. Colella and P.R. Woodward, *The piecewise parabolic method (PPM) for gas-dynamical simulations*, Journal of Computational Physics 54 (1984) 174-201

[25] A. Dedner, F. Kemm, D. Kröener, C.-D. Munz, T. Schnitzer, M. Wesenberg, *Hyperbolic divergence cleaning for MHD equations*, Journal of Computational Physics 175 (2002) 645-673

[26] M., Dumbser, M., Käser, *Arbitary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems*, Journal of Computational Physics, 221 (2007) 693-723

[27] Dumbser, M., Balsara, D.S., Toro, E.F., Munz, C.-D., *A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes*, Journal of Computational Physics, 227 (2008) 8209-8253

[28] M. Dumbser and E. F. Toro. *A simple extension of the Osher Riemann solver to non-conservative hyperbolic systems*, Journal of Scientific Computing, 48:70 (2011)88

[29] B.Einfeldt, *On Godunov-type methods for gas dynamics*, SIAM J. Numer. Anal., 25(3) (1988) 294-318

[30] B.Einfeldt, C.-D. Munz, P.L. Roe & B. Sjogreen, *On Godunov-type methods near low densities*, J. Comput. Phys., 92 (1991) 273-295

[31] M. Fey, *Multidimensional upwinding 1. The method of transport for solving the Euler equations*, J. Comput. Phys., 143 (1998) 159

[32] M. Fey, *Multidimensional upwinding 2. Decomposition of the Euler equation into advection equation,* J. Comput. Phys., 143 (1998) 181

[33] H. Gilquin, J. Laurens, and C. Rosier, *Multidimensional Riemann problems for linear hyperbolic system*s, Notes Numer. Fluid Mech., 43 (1993) 284

[34] S.K. Godunov, *Finite Difference Methods for the Computation of Discontinuous Solutions of the Equations of Fluid Dynamics*, Mathematics of the USSR, Sbornik. 47 (1959) 271-306

[35] S.K. Godunov, *Numerical Solution of Multi-dimensional Problems in Gas Dynamics*, Nauka Press, Moscow (1976)

[36] K.F. Gurski, *An HLLC-type approximate Riemann solver for ideal magnetohydrodynamics*, SIAM J. Sci. Comput. 25 (2004) 2165

[37] A. Harten, P.D. Lax and B. van Leer, *On upstream differencing and Godunov-type schemes for hyperbolic conservation laws*, SIAM Rev. (1983) 25, 289-315

[38] Jiang, G.-S. and Shu, C.-W., *Efficient implementation of weighted ENO schemes*, Journal of Computational Physics, 126 (1996) 202-228

[39] R.J. LeVeque, *Wave propagation algorithms for multidimensional hyperbolic systems*, J. Comput. Phys. 131 (1997) 327

[40] S.-T. Li, *An HLLC Riemann solver for magnetohydrodynamics*, J. Comput. Phys., 203 (2005) 344

[41] M. Lukacsova-Medvidova, K.W. Morton, G. Warnecke, *Finite volume evolution Galerkin methods for Euler equations of gas dynamics*, Int. J. Num. Methods in Fluids, 40 (2002) 425

[42] P.-H. Maire, R. Abgrall, J. Breil, J. Ovadia, A cell-centered Lagrangian scheme for two-dimensional compressible flow problems, SIAM J. Sci. Comput. 29(4) (2007) 1781–1824

[43] T. Miyoshi and K. Kusano, *A multi-state HLL approximate Riemann solver for ideal magnetohydrodynamics*, J. Comput. Phys., 208 (2005) 315-344

[44] S. A. Orszag and C. M. Tang, *Small-scale structure of two-dimensional magnetohydrodynamic turbulence*, Journal of Fluid Mechanics, 90 (1979) 129

[45] S. Osher and F. Solomon, *Upwind Difference Schemes for Hyperbolic Systems of Conservation Laws*, Mathematics of Computation, 38(158) (1982) 339

[46] P.L. Roe, *Approximate Riemann solver, parameter vectors and difference schemes*, Journal of Computational Physics 43 (1981) 357-372

[47] P.L. Roe, *Discrete models for the numerical analysis of time-dependent multidimensional gas dynamics*, J. Comput. Phys., 63 (1986) 458

[48] P. L. Roe and D. S. Balsara, *Notes on the eigensystem of magnetohydrodynamics*, SIAM Journal of applied Mathematics 56 (1996), 57

[49] C.B. Rumsey, B. van Leer & P.L. Roe, *A multidimensional flux function with application to the Euler and Navier-Stokes equations*, J. Comput. Phys., 105 (1993) 306

[50] V.V. Rusanov, Calculation of interaction of non-steady shock waves with obstacles, J. Comput. Math. Phys. USSR, 1 (1961) 267

[51] J. Saltzman, An *Unsplit 3D Upwind Method for Hyperbolic Conservation Laws*, J. Comput. Phys., 115 (1994) 153

[52] C.W. Schulz-Rinne, J.P. Collins, and H.M. Glaz, *Numerical solution of the Riemann problem for two-dimensional gas dynamics*, SIAM J. Sci. Comput., 14(7) (1993) 1394-1414

[53] Titarev, V.A. and Toro, E.F., *ADER: arbitrary high order Godunov approach*, Journal of Scientific Computing 17 (1-4) (2002) 609-618

[54] Titarev, V.A. and Toro, E.F., *ADER schemes for three-dimensional nonlinear hyperbolic systems*, Journal of Computational Physics, 204 (2005) 715-736

[55] Toro, E.F. and Titarev, V.A., *Solution of the generalized Riemann problem for advection reaction equations*, Proceedings of the Royal Society of London, Series A 458 (2002) 271-281

[56] E.F. Toro, M. Spruce and W. Speares, *Restoration of contact surface in the HLL Riemann solver*, Shock Waves, 4 (1994) 25-34

[57] E F Toro, M Spruce and W Speares. *Restoration of the contact surface in the Harten-Lax-van Leer Riemann solver*. Shock Waves. Vol. 4 (1994) pages 25-34

[58] E F Toro, M Spruce and W Speares, *Restoration of the contact surface in the HLL Riemann solver*, Technical report CoA 9204. Department of Aerospace Science, College of Aeronautics, Cranfield Institute of Technology. UK. June, 1992

[59] B. van Leer, *Toward the Ultimate Conservative Difference Scheme. V. A Second-Order Sequel to Godunov's Method*, J. Comput. Phys., 32 (1979) 101

[60] B. Wendroff, *A two-dimensional HLLE Riemann solver and associated Godunov-type difference scheme for gas dynamics*, Computers and Mathematics with Applications, 38 (1999) 175-185

[61] P. Woodward and P. Colella, *The numerical simulation of two-dimensional fluid flow with strong shocks*, Journal of Computational Physics 54 (1984), 115-173

Appendix A

In this Appendix we focus on the construction of the one-dimensional HLLC Riemann solver directly in rotated coordinates with a minimal number of coordinate transformations. We also explain how we can obtain the fluxes at each of the faces that meet at a vertex. The emphasis is on finding the one-dimensional fluxes in the longitudinal and transverse directions at each face. This is illustrated for the Euler equations.

To build the multi-dimensional Riemann solver, we have to solve for the one-dimensional HLLC fluxes (along the facial normal and orthogonal to it). To that end, let (x, y) be coordinates in a frame with $(\mathbf{x}, \mathbf{y})$ as the basis vectors. This is the frame in which we are solving the global problem. Let (η, τ) be coordinates in a rotated frame with $(\boldsymbol{\eta}, \boldsymbol{\tau})$ as the basis vectors. The latter frame of reference is rotated relative to the global frame of reference. We have $\boldsymbol{\eta} = \eta_x \mathbf{x} + \eta_y \mathbf{y}$ and $\boldsymbol{\tau} = -\eta_y \mathbf{x} + \eta_x \mathbf{y}$. In this discussion, $\boldsymbol{\eta}$ will eventually be equated to one of the unit normals $\boldsymbol{\eta}_i$, and $\boldsymbol{\tau}$ will be equated to the corresponding unit vector $\boldsymbol{\tau}_i$. The transcription of vectors and coordinates from the (η, τ) frame to the (x, y) frame is given by

$$\mathrm{v}_x = \eta_x \ \mathrm{v}_\eta - \eta_y \ \mathrm{v}_\tau \quad ; \quad \mathrm{v}_y = \eta_y \ \mathrm{v}_\eta + \eta_x \ \mathrm{v}_\tau \quad ; \quad x = \eta_x \ \eta - \eta_y \ \tau \quad ; \quad y = \eta_y \ \eta + \eta_x \ \tau \tag{A.1}$$

The reverse transcription of vectors and coordinates from the (x, y) frame to the (η, τ) frame is given by

$$\mathrm{v}_\eta = \eta_x \ \mathrm{v}_x + \eta_y \ \mathrm{v}_y \quad ; \quad \mathrm{v}_\tau = -\eta_y \ \mathrm{v}_x + \eta_x \ \mathrm{v}_y \quad ; \quad \eta = \eta_x \ x + \eta_y \ y \quad ; \quad \tau = -\eta_y \ x + \eta_x \ y \tag{A.2}$$

The partial derivatives also get transcribed as

$$\frac{\partial}{\partial x} = \eta_x \ \frac{\partial}{\partial \eta} - \eta_y \ \frac{\partial}{\partial \tau} \quad ; \quad \frac{\partial}{\partial y} = \eta_y \ \frac{\partial}{\partial \eta} + \eta_x \ \frac{\partial}{\partial \tau} \tag{A.3}$$

and

$$\frac{\partial}{\partial \eta} = \eta_x \ \frac{\partial}{\partial x} + \eta_y \ \frac{\partial}{\partial y} \quad ; \quad \frac{\partial}{\partial \tau} = -\eta_y \ \frac{\partial}{\partial x} + \eta_x \ \frac{\partial}{\partial y} \tag{A.4}$$

This completes the process of describing how the coordinate transformation works.

Let us now consider the Euler equations in the global frame:

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho\,\mathrm{v}_x\\ \rho\,\mathrm{v}_y\\ \rho\,\mathrm{v}_z\\ \mathcal{E}\end{pmatrix}+\frac{\partial}{\partial x}\begin{pmatrix}\rho\,\mathrm{v}_x\\ \rho\,\mathrm{v}_x\,\mathrm{v}_x+\mathrm{P}\\ \rho\,\mathrm{v}_y\,\mathrm{v}_x\\ \rho\,\mathrm{v}_z\,\mathrm{v}_x\\ (\mathcal{E}+\mathrm{P})\,\mathrm{v}_x\end{pmatrix}+\frac{\partial}{\partial y}\begin{pmatrix}\rho\,\mathrm{v}_y\\ \rho\,\mathrm{v}_x\,\mathrm{v}_y\\ \rho\,\mathrm{v}_y\,\mathrm{v}_y+\mathrm{P}\\ \rho\,\mathrm{v}_z\,\mathrm{v}_y\\ (\mathcal{E}+\mathrm{P})\,\mathrm{v}_y\end{pmatrix}+\frac{\partial}{\partial z}\begin{pmatrix}\rho\,\mathrm{v}_z\\ \rho\,\mathrm{v}_x\,\mathrm{v}_z\\ \rho\,\mathrm{v}_y\,\mathrm{v}_z\\ \rho\,\mathrm{v}_z\,\mathrm{v}_z+\mathrm{P}\\ (\mathcal{E}+\mathrm{P})\,\mathrm{v}_z\end{pmatrix}=0 \tag{A.5}$$

In the rotated frame, we have

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho\,\mathrm{v}_x\\ \rho\,\mathrm{v}_y\\ \rho\,\mathrm{v}_z\\ \mathcal{E}\end{pmatrix}+\frac{\partial}{\partial \eta}\begin{pmatrix}\rho\,\mathrm{v}_\eta\\ \rho\,\mathrm{v}_x\,\mathrm{v}_\eta+\eta_x\,\mathrm{P}\\ \rho\,\mathrm{v}_y\,\mathrm{v}_\eta+\eta_y\,\mathrm{P}\\ \rho\,\mathrm{v}_z\,\mathrm{v}_\eta\\ (\mathcal{E}+\mathrm{P})\,\mathrm{v}_\eta\end{pmatrix}+\frac{\partial}{\partial \tau}\begin{pmatrix}\rho\,\mathrm{v}_\tau\\ \rho\,\mathrm{v}_x\,\mathrm{v}_\tau-\eta_y\,\mathrm{P}\\ \rho\,\mathrm{v}_y\,\mathrm{v}_\tau+\eta_x\,\mathrm{P}\\ \rho\,\mathrm{v}_z\,\mathrm{v}_\tau\\ (\mathcal{E}+\mathrm{P})\,\mathrm{v}_\tau\end{pmatrix}+\frac{\partial}{\partial z}\begin{pmatrix}\rho\,\mathrm{v}_z\\ \rho\,\mathrm{v}_x\,\mathrm{v}_z\\ \rho\,\mathrm{v}_y\,\mathrm{v}_z\\ \rho\,\mathrm{v}_z\,\mathrm{v}_z+\mathrm{P}\\ (\mathcal{E}+\mathrm{P})\,\mathrm{v}_z\end{pmatrix}=0 \tag{A.6}$$

The definitions of $\mathrm{v}_\eta=\eta_x\,\mathrm{v}_x+\eta_y\,\mathrm{v}_y$ and $\mathrm{v}_\tau=-\eta_y\,\mathrm{v}_x+\eta_x\,\mathrm{v}_y$ are as given previously. Now notice from the previous narrative that we want to evaluate the one-dimensional HLLC Riemann solver in the direction $\boldsymbol{\tau}$. A first step towards that goal would be to obtain the one-dimensional HLL Riemann solver. Thus we focus on the part of the conservation law that is given by

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho\,\mathrm{v}_x\\ \rho\,\mathrm{v}_y\\ \rho\,\mathrm{v}_z\\ \mathcal{E}\end{pmatrix}+\frac{\partial}{\partial \tau}\begin{pmatrix}\rho\,\mathrm{v}_\tau\\ \rho\,\mathrm{v}_x\,\mathrm{v}_\tau-\eta_y\,\mathrm{P}\\ \rho\,\mathrm{v}_y\,\mathrm{v}_\tau+\eta_x\,\mathrm{P}\\ \rho\,\mathrm{v}_z\,\mathrm{v}_\tau\\ (\mathcal{E}+\mathrm{P})\,\mathrm{v}_\tau\end{pmatrix}=0 \quad\Leftrightarrow\quad \partial_t\mathbf{U}+\partial_\tau\mathbf{F}_{\boldsymbol{\tau}}=0 \tag{A.7}$$

The above equation serves to define $\mathbf{F}_{\boldsymbol{\tau}}=-\eta_y\mathbf{F}+\eta_x\mathbf{G}$ and $\mathbf{G}_{\boldsymbol{\eta}}=\eta_x\mathbf{F}+\eta_y\mathbf{G}$ as the Euler fluxes in the mutually orthogonal $\boldsymbol{\tau}$ and $\boldsymbol{\eta}$ directions. Let $S_{\boldsymbol{\tau}}^+$ and $S_{\boldsymbol{\tau}}^-$ be the extremal wave speeds along the direction $\boldsymbol{\tau}$. Let $\mathbf{U}_L$ and $\mathbf{U}_R$ be the left and right conserved quantities and let $\mathbf{F}_{\boldsymbol{\tau};L}$ and $\mathbf{F}_{\boldsymbol{\tau};R}$ be the corresponding fluxes. Let $\mathbf{U}^*$ and $\mathbf{F}_{\boldsymbol{\tau}}^*$ be the intermediate state and τ-directional flux produced by the one-dimensional HLL Riemann solver. We then have:

$$\mathbf{U}^*=\frac{S_{\boldsymbol{\tau}}^+\,\mathbf{U}_R-S_{\boldsymbol{\tau}}^-\,\mathbf{U}_L-\left(\mathbf{F}_{\boldsymbol{\tau};R}-\mathbf{F}_{\boldsymbol{\tau};L}\right)}{S_{\boldsymbol{\tau}}^+-S_{\boldsymbol{\tau}}^-} \tag{A.8}$$

and

$$\mathbf{F}_{\boldsymbol{\tau}}^*=\left[\frac{S_{\boldsymbol{\tau}}^+}{S_{\boldsymbol{\tau}}^+-S_{\boldsymbol{\tau}}^-}\right]\mathbf{F}_{\boldsymbol{\tau};L}-\left[\frac{S_{\boldsymbol{\tau}}^-}{S_{\boldsymbol{\tau}}^+-S_{\boldsymbol{\tau}}^-}\right]\mathbf{F}_{\boldsymbol{\tau};R}+\left[\frac{S_{\boldsymbol{\tau}}^+\,S_{\boldsymbol{\tau}}^-}{S_{\boldsymbol{\tau}}^+-S_{\boldsymbol{\tau}}^-}\right]\left(\mathbf{U}_R-\mathbf{U}_L\right) \tag{A.9}$$

This completes the process of describing how the one-dimensional HLL flux is obtained.

Once the intermediate state and flux of the one-dimensional HLL Riemann solver are obtained, we can write the physical variables in the intermediate state as:

$$\rho^* = \left(\mathbf{U}^*\right)_1 \;\; ; \;\; \mathrm{v}_x^* = \frac{\left(\mathbf{U}^*\right)_2}{\left(\mathbf{U}^*\right)_1} \;\; ; \;\; \mathrm{v}_y^* = \frac{\left(\mathbf{U}^*\right)_3}{\left(\mathbf{U}^*\right)_1} \;\; ; \;\; \mathrm{v}_z^* = \frac{\left(\mathbf{U}^*\right)_4}{\left(\mathbf{U}^*\right)_1} \;\; ;$$

$$S_{\boldsymbol{\tau}}^M \equiv \mathrm{v}_\tau^* \equiv -\eta_y \, \mathrm{v}_x^* + \eta_x \, \mathrm{v}_y^* \;\; ; \;\; \mathrm{v}_\eta^* \equiv \eta_x \, \mathrm{v}_x^* + \eta_y \, \mathrm{v}_y^* \;\; ;$$

$$\mathrm{P}_T^* \equiv \frac{1}{2}\Big[\mathrm{P}_L + \mathrm{P}_R + \rho_L \left(S_{\boldsymbol{\tau}}^- - \mathrm{v}_{\tau L}\right)\left(S_{\boldsymbol{\tau}}^M - \mathrm{v}_{\tau L}\right) + \rho_R \left(S_{\boldsymbol{\tau}}^+ - \mathrm{v}_{\tau R}\right)\left(S_{\boldsymbol{\tau}}^M - \mathrm{v}_{\tau R}\right)\Big] \tag{A.10}$$

Here P_T^* is the gas pressure. The speeds $\mathrm{v}_\eta^* + \sqrt{\Gamma \, \mathrm{P}_T^* / \rho^*}$ and $\mathrm{v}_\eta^* - \sqrt{\Gamma \, \mathrm{P}_T^* / \rho^*}$ can be used to further increase the extent of the maximal speed along the unit vector $\boldsymbol{\eta}$. This could give better multidimensional stability. Once the intermediate pressure, P^*, and velocity, $S_{\boldsymbol{\tau}}^M$, are found, they enable us to linearize the longitudinal and transverse flux in terms of the intermediate state variables. Thus we have two states $\mathbf{U}^{*+}$ and $\mathbf{U}^{*-}$. The corresponding fluxes in the $\boldsymbol{\tau}$ direction are given by $\mathbf{F}_{\boldsymbol{\tau}}^{*+}$ and $\mathbf{F}_{\boldsymbol{\tau}}^{*-}$. In the $\boldsymbol{\eta}$ direction, the fluxes are given by $\mathbf{G}_{\boldsymbol{\eta}}^{*+}$ and $\mathbf{G}_{\boldsymbol{\eta}}^{*-}$. We now have:

$$\mathbf{F}_{\boldsymbol{\tau}}^{*+} = S_{\boldsymbol{\tau}}^M \; \mathbf{U}^{*+} + \left(0, \;\; -\eta_y \, \mathrm{P}^*, \;\; \eta_x \, \mathrm{P}^*, \;\; 0, \;\; \mathrm{P}^* S_{\boldsymbol{\tau}}^M\right)^T \;\; ;$$

$$\mathbf{F}_{\boldsymbol{\tau}}^{*-} = S_{\boldsymbol{\tau}}^M \; \mathbf{U}^{*-} + \left(0, \;\; -\eta_y \, \mathrm{P}^*, \;\; \eta_x \, \mathrm{P}^*, \;\; 0, \;\; \mathrm{P}^* S_{\boldsymbol{\tau}}^M\right)^T \;\; ;$$

$$\mathbf{G}_{\boldsymbol{\eta}}^{*+} = \mathrm{v}_\eta^* \; \mathbf{U}^{*+} + \left(0, \;\; \eta_x \, \mathrm{P}^*, \;\; \eta_y \, \mathrm{P}^*, \;\; 0, \;\; \mathrm{P}^* \mathrm{v}_\eta^*\right)^T \;\; ;$$

$$\mathbf{G}_{\boldsymbol{\eta}}^{*-} = \mathrm{v}_\eta^* \; \mathbf{U}^{*-} + \left(0, \;\; \eta_x \, \mathrm{P}^*, \;\; \eta_y \, \mathrm{P}^*, \;\; 0, \;\; \mathrm{P}^* \mathrm{v}_\eta^*\right)^T \tag{A.11}$$

The above fluxes will be fully specified once we have the two states $\mathbf{U}^{*+}$ and $\mathbf{U}^{*-}$. This is done by writing the shock jump conditions across the two fastest propagating waves in the problem. We get:

$$\mathbf{F}_{\boldsymbol{\tau}}^{*+} - S_{\boldsymbol{\tau}}^+ \; \mathbf{U}^{*+} = \mathbf{F}_{\boldsymbol{\tau};R} - S_{\boldsymbol{\tau}}^+ \; \mathbf{U}_R \;\; ; \;\; \mathbf{F}_{\boldsymbol{\tau}}^{*-} - S_{\boldsymbol{\tau}}^- \; \mathbf{U}^{*-} = \mathbf{F}_{\boldsymbol{\tau};L} - S_{\boldsymbol{\tau}}^- \; \mathbf{U}_L \tag{A.12}$$

The final closed-form expressions for the two states $\mathbf{U}^{*+}$ and $\mathbf{U}^{*-}$ are easily obtained as:

$$\mathbf{U}^{*+} = \frac{\mathbf{F}_{\boldsymbol{\tau};R} - S_{\boldsymbol{\tau}}^+ \; \mathbf{U}_R - \left(0, \;\; -\eta_y \, \mathrm{P}^*, \;\; \eta_x \, \mathrm{P}^*, \;\; 0, \;\; \mathrm{P}^* S_{\boldsymbol{\tau}}^M\right)^T}{S_{\boldsymbol{\tau}}^M - S_{\boldsymbol{\tau}}^+}$$

$$\mathbf{U}^{*-} = \frac{\mathbf{F}_{\boldsymbol{\tau};L} - S_{\boldsymbol{\tau}}^- \; \mathbf{U}_L - \left(0, \;\; -\eta_y \, \mathrm{P}^*, \;\; \eta_x \, \mathrm{P}^*, \;\; 0, \;\; \mathrm{P}^* S_{\boldsymbol{\tau}}^M\right)^T}{S_{\boldsymbol{\tau}}^M - S_{\boldsymbol{\tau}}^-} \tag{A.13}$$

Once the two states, $\mathbf{U}^{*+}$ and $\mathbf{U}^{*-}$, as well as the four fluxes, $\mathbf{F}_{\boldsymbol{\tau}}^{*+}$, $\mathbf{F}_{\boldsymbol{\tau}}^{*-}$, $\mathbf{G}_{\boldsymbol{\eta}}^{*+}$ and $\mathbf{G}_{\boldsymbol{\eta}}^{*-}$, are obtained, we can obtain the fluxes in the global frame as:

$$\begin{aligned}
&\mathbf{F}^{*+} \equiv \eta_x \mathbf{G}_{\boldsymbol{\eta}}^{*+} - \eta_y \mathbf{F}_{\boldsymbol{\tau}}^{*+} \quad ; \quad \mathbf{F}^{*-} \equiv \eta_x \mathbf{G}_{\boldsymbol{\eta}}^{*-} - \eta_y \mathbf{F}_{\boldsymbol{\tau}}^{*-} \quad ; \\
&\mathbf{G}^{*+} \equiv \eta_y \mathbf{G}_{\boldsymbol{\eta}}^{*+} + \eta_x \mathbf{F}_{\boldsymbol{\tau}}^{*+} \quad ; \quad \mathbf{G}^{*-} \equiv \eta_y \mathbf{G}_{\boldsymbol{\eta}}^{*-} + \eta_x \mathbf{F}_{\boldsymbol{\tau}}^{*-}
\end{aligned} \tag{A.14}$$

This completes our description of the one-dimensional HLLC Riemann solver for Euler flow.

Appendix B

This Appendix explains the process of obtaining the wave model. An essential step in that process consists of obtaining the maximal speeds S_i along the unit normals $\boldsymbol{\eta}_i$.

The multidimensional wave model is obtained via an iterative process. Two variants are described here. In the first variant, the strongly interacting state will not contain the time axis in the supersonic limit, though it will contain the time axis in the subsonic limit. Consequently, in the supersonic limit, one has to resort to the procedure described in sub-section II.3.d. In the second variant, the strongly interacting state will always contain the time axis and all the fluxes are always obtained from the subsonic variant. The procedure described in sub-section II.3.d becomes unnecessary. The second variant is a little easier to implement and it is also a little more stable. The entire wave model can be built by following the three steps described below. The first and second steps are done only once at the start. The third step is iterated to convergence.

First Step:

Expand the ellipses and their bounding rectangles in Fig. 3b by just a little bit:

$$\begin{aligned}
&\Lambda_{\boldsymbol{\eta}i}^{+} \equiv \frac{1}{2}(1-\delta)\left(S_{\boldsymbol{\eta}i}^{+} + S_{\boldsymbol{\eta}i}^{-}\right) + \delta S_{\boldsymbol{\eta}i}^{+} \quad ; \quad \Lambda_{\boldsymbol{\eta}i}^{-} \equiv \frac{1}{2}(1-\delta)\left(S_{\boldsymbol{\eta}i}^{+} + S_{\boldsymbol{\eta}i}^{-}\right) + \delta S_{\boldsymbol{\eta}i}^{-} \quad ; \\
&\Lambda_{\boldsymbol{\tau}j}^{+} \equiv \frac{1}{2}(1-\delta)\left(S_{\boldsymbol{\tau}j}^{+} + S_{\boldsymbol{\tau}j}^{-}\right) + \delta S_{\boldsymbol{\tau}j}^{+} \quad ; \quad \Lambda_{\boldsymbol{\tau}j}^{-} \equiv \frac{1}{2}(1-\delta)\left(S_{\boldsymbol{\tau}j}^{+} + S_{\boldsymbol{\tau}j}^{-}\right) + \delta S_{\boldsymbol{\tau}j}^{-}
\end{aligned} \tag{B.1}$$

A reasonable choice would be to use $\delta =$ 1.02 to 1.05 in the above formulae. We also obtain the maximal speed in our wave model, $\Lambda_{\max}$, as:

$$\Lambda_{\max} = \max\left\{ \max\left(\sqrt{\left(S_{\boldsymbol{\eta}i}^{+}\right)^2 + \left(S_{\boldsymbol{\tau}j}^{+}\right)^2},\ \sqrt{\left(S_{\boldsymbol{\eta}i}^{+}\right)^2 + \left(S_{\boldsymbol{\tau}j}^{-}\right)^2},\ \sqrt{\left(S_{\boldsymbol{\eta}i}^{-}\right)^2 + \left(S_{\boldsymbol{\tau}j}^{+}\right)^2},\ \sqrt{\left(S_{\boldsymbol{\eta}i}^{-}\right)^2 + \left(S_{\boldsymbol{\tau}j}^{-}\right)^2} \right) \mid \mathrm{i} = 1, \mathit{imax} \right\} \tag{B.2}$$

The two different variants of wave model that we describe here are entirely determined by the value of that we start the iteration with:

$$\begin{aligned} S_i &= -100\ \Lambda_{\max} \qquad \text{for Variant \# 1} \\ S_i &= \left(\delta - 1\right)\ \Lambda_{\max} \qquad \text{for Variant \# 2} \end{aligned} \tag{B.3}$$

We see that the second choice compels the wave model that we construct to be subsonic. (It is also worth mentioning that one can build a multidimensional LLF Riemann solver by setting $S_i = \Lambda_{\max}$.)

Second Step:

The four vertices of each expanded rectangle R_j in Fig. 3b are given by $\Lambda^+_{\boldsymbol{\eta} j}\ \boldsymbol{\eta}_j + \Lambda^+_{\boldsymbol{\tau} j}\ \boldsymbol{\tau}_j$, $\Lambda^+_{\boldsymbol{\eta} j}\ \boldsymbol{\eta}_j + \Lambda^-_{\boldsymbol{\tau} j}\ \boldsymbol{\tau}_j$, $\Lambda^-_{\boldsymbol{\eta} j}\ \boldsymbol{\eta}_j + \Lambda^+_{\boldsymbol{\tau} j}\ \boldsymbol{\tau}_j$ and $\Lambda^-_{\boldsymbol{\eta} j}\ \boldsymbol{\eta}_j + \Lambda^-_{\boldsymbol{\tau} j}\ \boldsymbol{\tau}_j$. They can be projected on to the unit vector $\boldsymbol{\eta}_i$ to get a measure of the maximal speed contributed by the $\left(\eta_j, \tau_j\right)$ coordinate system in the $\boldsymbol{\eta}_i$ direction. Thus we have

$$\Lambda_{j\to i} = \max \begin{pmatrix} \Lambda^+_{\boldsymbol{\eta} j}\ \left(\boldsymbol{\eta}_i \bullet \boldsymbol{\eta}_j\right) + \Lambda^+_{\boldsymbol{\tau} j}\ \left(\boldsymbol{\eta}_i \bullet \boldsymbol{\tau}_j\right)\ ,\ \Lambda^+_{\boldsymbol{\eta} j}\ \left(\boldsymbol{\eta}_i \bullet \boldsymbol{\eta}_j\right) + \Lambda^-_{\boldsymbol{\tau} j}\ \left(\boldsymbol{\eta}_i \bullet \boldsymbol{\tau}_j\right)\ , \\ \Lambda^-_{\boldsymbol{\eta} j}\ \left(\boldsymbol{\eta}_i \bullet \boldsymbol{\eta}_j\right) + \Lambda^+_{\boldsymbol{\tau} j}\ \left(\boldsymbol{\eta}_i \bullet \boldsymbol{\tau}_j\right)\ ,\ \Lambda^-_{\boldsymbol{\eta} j}\ \left(\boldsymbol{\eta}_i \bullet \boldsymbol{\eta}_j\right) + \Lambda^-_{\boldsymbol{\tau} j}\ \left(\boldsymbol{\eta}_i \bullet \boldsymbol{\tau}_j\right) \end{pmatrix} \tag{B.4}$$

Please note that eqn. (B.4) is not a matrix but rather just a list of four numbers whose maximum we wish to take. We will get contributions like the one above from all the different $\left(\eta_j, \tau_j\right)$ coordinate systems in which one-dimensional Riemann problems were solved. We maximize all those contributions in order to get the largest speed along the unit vector $\boldsymbol{\eta}_i$, which yields our first guess for S_i . Thus we have

$$S_i = \max\left(S_i, \Lambda_{j\to i}\right) \qquad \forall\ j = 1,\ imax \tag{B.5}$$

One ingredient of this process consists of finding all the $\Lambda_{j\to i}$ for all the "*j*" relative to a particular "*i*" using eqn. (B.4) and then applying eqn. (B.5). A complete iteration consists of doing this for all the zone boundaries "*i*" that come together at a vertex. We illustrate one iteration for the special case of Fig. 1a below.

To consider an example from Fig. 1a, say we want the speed S_1 along $\boldsymbol{\eta}_1$. The three coordinate systems that we will then have to visit are indeed $\left(\eta_1, \tau_1\right)$, $\left(\eta_2, \tau_2\right)$ and $\left(\eta_3, \tau_3\right)$. In each of these coordinate systems we have already solved a one-dimensional Riemann problem which gives us the extremal speeds in the longitudinal and transverse directions. We visit each of these three coordinate systems and use eqn. (B.4) to form $\Lambda_{1\to 1}$, $\Lambda_{2\to 1}$ and $\Lambda_{3\to 1}$. We also form $\Lambda_{1\to 2}$, $\Lambda_{2\to 2}$ and $\Lambda_{3\to 2}$. And we also form $\Lambda_{1\to 3}$, $\Lambda_{2\to 3}$ and $\Lambda_{3\to 3}$. We can then say that

$$
\begin{aligned}
S_1 &= \max\left(S_1,\ \Lambda_{1\to 1}\ ,\ \Lambda_{2\to 1}\ ,\ \Lambda_{3\to 1}\ \right)\\
S_2 &= \max\left(S_2,\ \Lambda_{1\to 2}\ ,\ \Lambda_{2\to 2}\ ,\ \Lambda_{3\to 2}\ \right)\\
S_3 &= \max\left(S_3,\ \Lambda_{1\to 3}\ ,\ \Lambda_{2\to 3}\ ,\ \Lambda_{3\to 3}\ \right)
\end{aligned}
\qquad \text{(B.6)}
$$

Eqn. (B.6) completes one iteration of this step for a wave model that forms a triangular pyramid.

Third Step:

Let S_j be one of the positive maximal speeds that we found in the first step along one of the principal directions $\boldsymbol{\eta}_j$. Orthogonal to $\boldsymbol{\eta}_j$, i.e. in the $\boldsymbol{\tau}_j$ direction, we indeed solve a one-directional Riemann problem with extremal speeds $S^-_{\boldsymbol{\tau}j}$ and $S^+_{\boldsymbol{\tau}j}$. For the self-similar structures in Figs. 1 and 2 to be relevant, the vectors $S_j\boldsymbol{\eta}_j + S^-_{\boldsymbol{\tau}j}\boldsymbol{\tau}_j$ and $S_j\boldsymbol{\eta}_j + S^+_{\boldsymbol{\tau}j}\boldsymbol{\tau}_j$ (as specified in the $\left(\eta_j, \tau_j\right)$ frame) should be entirely contained in the side panel of the j^{th} segment $\mathrm{P}_i\mathrm{P}_{cyc(i+1)}$ shown in Fig. 1. With $i \neq j$, let $\boldsymbol{\eta}_i$ be one of the other principal directions that come together at vertex "O". The wave front associated with $\boldsymbol{\eta}_i$ should be such that it does not intersect the j^{th} wavefront anywhere between the vectors $S_j\boldsymbol{\eta}_j + S^-_{\boldsymbol{\tau}j}\boldsymbol{\tau}_j$ and $S_j\boldsymbol{\eta}_j + S^+_{\boldsymbol{\tau}j}\boldsymbol{\tau}_j$. If it does, the speed S_i should be further increased so that it stops intersecting the wave model of the one-dimensional Riemann problem in the j^{th} wavefront.

The strategy for implementing such an algorithm is to visit all other principal directions $\boldsymbol{\eta}_j$ with $j \neq i$. When the two vectors $S_j\boldsymbol{\eta}_j + S^-_{\boldsymbol{\tau}j}\boldsymbol{\tau}_j$ and $S_j\boldsymbol{\eta}_j + S^+_{\boldsymbol{\tau}j}\boldsymbol{\tau}_j$ are projected in the $\boldsymbol{\eta}_i$ direction, we allow S_i to increase as follows

$$
S_i = \max\left(S_i\ ,\ S_j\left(\boldsymbol{\eta}_i \bullet \boldsymbol{\eta}_j\right) + S^-_{\boldsymbol{\tau}j}\left(\boldsymbol{\eta}_i \bullet \boldsymbol{\tau}_j\right)\ ,\ S_j\left(\boldsymbol{\eta}_i \bullet \boldsymbol{\eta}_j\right) + S^+_{\boldsymbol{\tau}j}\left(\boldsymbol{\eta}_i \bullet \boldsymbol{\tau}_j\right)\ \right) \qquad \forall\ j = 1,\ imax \qquad \text{(B.7)}
$$

Please not that this step is applied after the first step has been applied to all the wavefronts. The procedure described in eqn. (B.7) is done for all "*i*", with the result that we obtain the convex hull of subsonic wavefronts that fully covers the origin and respects the wave model in each of the one-dimensional Riemann problems.

This procedure described in eqn. (B.7) is first done for all "i". If all the wave speeds S_i have not converged, the procedure is repeated. In practice we find that all the wave speeds S_i converge within a few iterations and the convergence depends on the quality of the mesh. We have always obtained convergence of the wave model in less than "3 × *imax*" iterations. For most problems on most meshes, we have obtained convergence in less than "*imax*" iterations.

Appendix C

Recall that we always use Gauss' Law to turn volumetric integrals of the conservation law over two space and one time dimension into area integrals over the faces of that volume. The area integrations of the conservation law over the side panels, which we sometimes refer to as panel integration, deserves explanation. In Fig. 5 we pictorially showed how the area integral can be evaluated over a portion of the side panel of the inverted pyramid. This Appendix focuses on the integration of the conservation law over that side panel.

The portion of the side panel is picked to have a constant state and corresponding fluxes. As shown in Fig. 5, let $\Delta \mathrm{OP_aP_b}$ contain a constant state **U**, with the associated fluxes **F** and **G** as shown. In Fig. 5, we focus on the first side panel from either Fig. 2b or Fig. 7b. Along the segment $\mathrm{P_1P_2}$, let the point $\mathrm{P_a}$ be identified with a local coordinate l_a . Similarly, the point $\mathrm{P_b}$ is identified with l_b . Here, l_a and l_b have units of speed and we do require $l_a > l_b$. We can now write

$$\begin{aligned} \frac{1}{T}\overrightarrow{\mathrm{OP_a}} &= S_1\boldsymbol{\eta}_1 + l_a\boldsymbol{\tau}_1 + \mathbf{t} = \left(S_1\eta_{1x} - l_a\eta_{1y}\right)\mathbf{x} + \left(S_1\eta_{1y} + l_a\eta_{1x}\right)\mathbf{y} + \mathbf{t} \\ \frac{1}{T}\overrightarrow{\mathrm{OP_b}} &= S_1\boldsymbol{\eta}_1 + l_b\boldsymbol{\tau}_1 + \mathbf{t} = \left(S_1\eta_{1x} - l_b\eta_{1y}\right)\mathbf{x} + \left(S_1\eta_{1y} + l_b\eta_{1x}\right)\mathbf{y} + \mathbf{t} \end{aligned} \tag{C.1}$$

The outward pointing area vector associated with $\Delta \mathrm{OP_aP_b}$ is given by

$$\frac{1}{T^2}\overrightarrow{\mathrm{OP_a}} \times \overrightarrow{\mathrm{OP_b}} = \frac{1}{2}\left(l_a - l_b\right)\left[\eta_{1x}\mathbf{x} + \eta_{1y}\mathbf{y} - S_1\mathbf{t}\right] \tag{C.2}$$

The dot product of the above area vector with $\mathbf{Fx} + \mathbf{Gy} + \mathbf{Ut}$ yields the area integral of the conservation law over the side panel of the inverted pyramid as:

$$\frac{1}{T^2}\left(\text{Area integral over } \mathrm{OP_aP_b}\right) = \frac{1}{2}\left(l_a - l_b\right)\left[\eta_{1x}\mathbf{F} + \eta_{1y}\mathbf{G} - S_1\mathbf{U}\right] \tag{C.3}$$

In many of the integrals over the side panels, we generically refer to the flux $\eta_{1x}\mathbf{F} + \eta_{1y}\mathbf{G}$ as $\mathbf{G}_{\boldsymbol{\eta}}$. In such situations, we do explicitly mention that $\mathbf{G}_{\boldsymbol{\eta}}$ is the flux in the $\boldsymbol{\eta}_1$ direction, see also eqn. (2.2).

On some occasions we will want to start our integration from a point P_{start} that is identified with a local coordinate l_{start} . For the sake of computer implementation, we want a formula that works regardless of the relation between l_{start} and the interval $\left[l_b, l_a\right]$. In that case, the integral from eqn. (C.3) gets modified to become

$$\frac{1}{T^2}\left(\text{Area integral over } \mathrm{OP_aP}_{start}\right) = \frac{1}{2}\left(l_a - \max\left(l_b, l_{start}\right)\right)\mathrm{H}\left(l_a - l_{start}\right)\left[\eta_{1x}\mathbf{F} + \eta_{1y}\mathbf{G} - S_1\mathbf{U}\right] \tag{C.4}$$

Similarly, we may want to end our integration at a point P_{end} that is identified with a local coordinate l_{end} . Now, the integral from eqn. (C.3) gets modified to become

$$\frac{1}{T^2}\left(\text{Area integral over OP}_{end}\text{P}_\text{b}\right)=\frac{1}{2}\left(\min\left(l_a,l_{end}\right)-l_b\right)\text{H}\left(l_{end}-l_b\right)\left[\eta_{1x}\mathbf{F}+\eta_{1y}\mathbf{G}-S_1\mathbf{U}\right] \qquad \text{(C.5)}$$

Eqns. (C.3), (C.4) and (C.5) only pertain to situations where a single state is being considered. When applied to an entire side panel of the inverted pyramid, we have to sum over all the states under consideration.

Appendix D

Here we give the analog of Sub-section III.4 as it applies to the MHD equations. For the MHD equations, realize that the contact cannot have a variation in the transverse magnetic field. As a result, the longitudinal and transverse magnetic field will have to be obtained from the HLL Riemann solver. Let the vector of conserved variables be given by $\mathbf{U}^*_{C1}=\left(\rho^{*C1},\ \rho^{*C1}\text{v}_x^{*C1},\ \rho^{*C1}\text{v}_y^{*C1},\ \rho^{*C1}\text{v}_z^{*C1},\ \varepsilon^{*C1},\ \text{B}_x^*,\ \text{B}_y^*,\ \text{B}_z^*\right)^T$. After writing analogous expressions for the other state and the fluxes, we can write the analogue of eqn. (3.21). It is

$$\text{A}_{UHLLC}\begin{pmatrix}\rho^{*C1}\\ \rho^{*C1}\text{v}_\sigma^*\\ \rho^{*C1}\text{v}_\kappa^{*C1}\\ \rho^{*C1}\text{v}_z^{*C1}\\ \varepsilon^{*C1}\\ \text{B}_x^*\\ \text{B}_y^*\\ \text{B}_z^*\end{pmatrix}+\frac{1}{2}l_{UHLLC}\psi_{UHLLC}\begin{pmatrix}\rho^{*C1}\left(\text{v}_\sigma^*-S_\sigma^*\right)\\ \rho^{*C1}\text{v}_\sigma^*\left(\text{v}_\sigma^*-S_\sigma^*\right)+\text{P}_{Tot}^*\\ \rho^{*C1}\text{v}_\kappa^{*C1}\left(\text{v}_\sigma^*-S_\sigma^*\right)\\ \rho^{*C1}\text{v}_z^{*C1}\left(\text{v}_\sigma^*-S_\sigma^*\right)\\ \varepsilon^{*C1}\left(\text{v}_\sigma^*-S_\sigma^*\right)+\text{P}_{Tot}^*\text{v}_\sigma^*\\ \text{B}_x^*\ \text{v}_\sigma^*\ -\ \text{v}_x^*\ \text{B}_\sigma^*\\ \text{B}_y^*\ \text{v}_\sigma^*\ -\ \text{v}_y^*\ \text{B}_\sigma^*\\ \text{B}_z^*\ \text{v}_\sigma^*\ -\ \text{v}_\kappa^*\ \text{B}_\sigma^*\end{pmatrix}=\begin{pmatrix}r_1^{C1}\\ r_\sigma^{C1}+l_{UHLLC}\psi_{UHLLC}\text{B}_\sigma^*\text{B}_\sigma^*/8\pi\\ r_\kappa^{C1}+l_{UHLLC}\psi_{UHLLC}\text{B}_\sigma^*\text{B}_\kappa^*/8\pi\\ r_4^{C1}+l_{UHLLC}\psi_{UHLLC}\text{B}_\sigma^*\text{B}_z^*/8\pi\\ r_5^{C1}+l_{UHLLC}\psi_{UHLLC}\text{B}_\sigma^*\left(\mathbf{v}^*\cdot\mathbf{B}^*\right)_{HLL}/8\pi\\ r_6^{C1}\\ r_7^{C1}\\ r_8^{C1}\end{pmatrix} \qquad \text{(D.1)}$$

We can also write the analogue of eqn. (3.22).

$$\left(\mathrm{A}_{UHLL}-\mathrm{A}_{UHLLC}\right)\begin{pmatrix}\rho^{*C2}\\ \rho^{*C2}\mathrm{v}_\sigma^*\\ \rho^{*C2}\mathrm{v}_\kappa^{*C2}\\ \rho^{*C2}\mathrm{v}_z^{*C2}\\ \varepsilon^{*C2}\\ \mathrm{B}_x^*\\ \mathrm{B}_y^*\\ \mathrm{B}_z^*\end{pmatrix}-\frac{1}{2}l_{UHLLC}\psi_{UHLLC}\begin{pmatrix}\rho^{*C2}\left(\mathrm{v}_\sigma^*-S_\sigma^*\right)\\ \rho^{*C2}\mathrm{v}_\sigma^*\left(\mathrm{v}_\sigma^*-S_\sigma^*\right)+\mathrm{P}_{Tot}^*\\ \rho^{*C2}\mathrm{v}_\kappa^{*C2}\left(\mathrm{v}_\sigma^*-S_\sigma^*\right)\\ \rho^{*C2}\mathrm{v}_z^{*C2}\left(\mathrm{v}_\sigma^*-S_\sigma^*\right)\\ \varepsilon^{*C2}\left(\mathrm{v}_\sigma^*-S_\sigma^*\right)+\mathrm{P}_{Tot}^*\mathrm{v}_\sigma^*\\ \mathrm{B}_x^*\,\mathrm{v}_\sigma^*-\mathrm{v}_x^*\,\mathrm{B}_\sigma^*\\ \mathrm{B}_y^*\,\mathrm{v}_\sigma^*-\mathrm{v}_y^*\,\mathrm{B}_\sigma^*\\ \mathrm{B}_z^*\,\mathrm{v}_\sigma^*-\mathrm{v}_\kappa^*\,\mathrm{B}_\sigma^*\end{pmatrix}=\begin{pmatrix}r_1^{C2}\\ r_\sigma^{C2}-l_{UHLLC}\psi_{UHLLC}\mathrm{B}_\sigma^*\mathrm{B}_\sigma^*/8\pi\\ r_\kappa^{C2}-l_{UHLLC}\psi_{UHLLC}\mathrm{B}_\sigma^*\mathrm{B}_\kappa^*/8\pi\\ r_4^{C2}-l_{UHLLC}\psi_{UHLLC}\mathrm{B}_\sigma^*\mathrm{B}_z^*/8\pi\\ r_5^{C2}-l_{UHLLC}\psi_{UHLLC}\mathrm{B}_\sigma^*\left(\mathbf{v}^*\cdot\mathbf{B}^*\right)_{HLL}/8\pi\\ r_6^{C2}\\ r_7^{C2}\\ r_8^{C2}\end{pmatrix} \tag{D.2}$$

Following Li [40] we notice that the magnetic field and magnetic flux components should be kept identical to their two-dimensional HLL counterparts on either side of the contact discontinuity. Furthermore, $\left(\mathbf{v}^*\cdot\mathbf{B}^*\right)_{HLL}$ should be obtained from the two-dimensional HLL Riemann solver. These choices are mandated by consistency considerations. The above way of writing the MHD equations on either side of the contact allow us to cast the system in a form that is closest to the Euler system. The analogue of eqn. (3.23) becomes

$$\begin{aligned}&\frac{1}{2}l_{UHLLC}\psi_{UHLLC}\mathrm{P}_{Tot}^*+r_1^{C1}\mathrm{v}_\sigma^*=r_\sigma^{C1}+l_{UHLLC}\psi_{UHLLC}\mathrm{B}_\sigma^*\mathrm{B}_\sigma^*/8\pi\\ &-\frac{1}{2}l_{UHLLC}\psi_{UHLLC}\mathrm{P}_{Tot}^*+r_1^{C2}\mathrm{v}_\sigma^*=r_\sigma^{C2}-l_{UHLLC}\psi_{UHLLC}\mathrm{B}_\sigma^*\mathrm{B}_\sigma^*/8\pi\end{aligned} \tag{D.3}$$

The above system of linear equations yields P_{Tot}^* and v_σ^*. The first rows of eqns. (D.1) and (D.2) can be used to obtain the densities ρ^{*C1} and ρ^{*C2}. The third and fourth rows of eqns. (D.1) and (D.2) can be used to obtain the transverse velocities v_κ^{*C1}, v_κ^{*C2}, v_z^{*C1} and v_z^{*C2}. The fifth rows of eqns. (D.1) and (D.2) can be used to obtain the total energies ε^{*C1} and ε^{*C2}. The rotation in eqn. (3.20) can be undone and the states $\mathbf{U}_{C1}^*$ and $\mathbf{U}_{C2}^*$ as well as the fluxes $\mathbf{F}_{C1}^*$, $\mathbf{G}_{C1}^*$, $\mathbf{F}_{C2}^*$ and $\mathbf{G}_{C2}^*$ can be assembled. The sixth through eighth components of the fluxes are reset to their corresponding HLL values in order to restore consistency with eqn. (3.24).

The fact that we were able to allow different transverse velocities but not different transverse magnetic fields in $\mathbf{U}_{C1}^*$ and $\mathbf{U}_{C2}^*$ highlights one of the inescapable deficiencies in the HLLC Riemann solver for MHD. This is a deficiency that can only be resolved with an HLLD-type Riemann solver.

Figure Captions

Fig. 1 shows the three states coming together at a vertex "O". The states are given by $\mathbf{U}_1$, $\mathbf{U}_2$ *and* $\mathbf{U}_3$. *They are separated by unit vectors* η_1, η_2 *and* η_3 *with the counterclockwise orientation shown. The fastest waves propagating along these unit vectors do so with speeds* S_1, S_2 *and* S_3 , *so that their location at a time "T" is shown by the grey lines. The intersection of these grey lines forms a triangle and the strongly interacting state is contained within that triangle.*

Fig. 2a shows the triangular prism in space and time that has the strongly interacting state $\mathbf{U}^*$ *for its top surface. The side panels of Fig. 2a depict the one-dimensional HLL Riemann problems. Imagine now that the side panels of Fig. 2a are removed to expose the space-time domain covered by the strongly interacting state. Fig. 2b shows the self-similar extent of that strongly interacting state in space-time. The projection of the one-dimensional HLL Riemann problems are also shown on the side surfaces of Fig. 2b. The triangle* OQ_1Q_2 *shows the surface used for obtaining the x-flux.*

Fig. 3a shows the rotated frame (h_i, t_i) *in which the shifted ellipse* E_i *is obtained. The rotated frame makes an angle q with the global (x,y) frame. Ellipse* E_i *represents the wave propagation in the rotated frame. The extent of the ellipse* E_i *in the rotated frame is also shown by the speeds "S". The unit vector in this frame along which we wish to find the maximal speed is shown as* $\mathbf{n}$. *It makes an angle f with the global (x,y) frame. Fig. 3b shows how the ellipse* E_i *in Fig. 3a is replaced with a bounding rotated rectangle* R_i . *The rectangle* R_i *circumscribes the ellipse and will, therefore, be a good proxy for the maximal speeds that can be represented by the ellipse. The vertices of the rectangle* R_i *are indeed easily specified in the rotated* (h_i, t_i) *frame. The projection of those vertices in the direction of the unit vector* $\mathbf{n}$ *gives the maximal speed contributed by* R_i *along* $\mathbf{n}$.

Fig. 4 shows a magnified view of the grey triangle from Fig. 1. The dashed lines demarcate the boundaries of the intermediate states associated with the one-dimensional HLL Riemann problems. The dashed lines are two-dimensional projections of the solid lines within the side panels of Fig. 2b. As depicted, the 1D Riemann problem is subsonic in faces 1 and 3 and supersonic in face 2. The orientation of the x- and y-axes is shown.

Fig. 5 shows an expanded version of triangle ΔOP_1P_2 *from Fig. 2b. It helps in illustrating the procedure for evaluating the area integral over a sub-triangle* $\Delta\ OP_aP_b$. *The state* $\mathbf{U}$ *and fluxes* $\mathbf{F}$ *and* $\mathbf{G}$ *are constant on this sub-triangle. The text shows how the facial integrals over sub-triangles are easily evaluated in various circumstances.*

Figs. 6a and 6b show the domains used for evaluating the x- and y-fluxes for the multidimensional HLL Riemann solver. A five-sided wave model is shown.

Figs. 6c and 6d show different propagation directions of the contact discontinuity for the multidimensional HLLC Riemann solver. The contact discontinuity is shown by the thick dashed line. Its orientation is given by the unit normal $\mathbf{s}$.

Fig. 7a shows the triangular prism in space and time that is used for obtaining the strongly interacting states for the multidimensional HLLC Riemann solver. The contact discontinuity, shown as a thick dashed line, separates the two strongly interacting states. Fig. 7b can be thought of as removing the side panels of Fig. 7a. The strongly-interacting states form an inverted triangular pyramid in space-time. The waves and states in Fig. 7b are only labeled in the face corresponding to η_1 .

Fig. 8 shows results from the Sod shock tube problem done on a 2D unstructured mesh. Fig. 8a shows the density and the mesh for a simulation that was run with the multidimensional HLLC Riemann solver with a CFL of 1; Fig. 8b shows the same when the 1D HLLC Riemann solver was used with the same CFL. Fig. 8c shows a plot of the density in one dimension for various simulations. Fig. 8b shows that the simulation goes unstable when a 1d Riemann solver is used with a CFL of unity.

Fig. 9 shows results from the Lax shock tube problem done on a 2D unstructured mesh. Fig. 9a shows the density and the mesh for a simulation that was run with the multidimensional HLLC Riemann solver with a CFL of 1. Fig. 9b shows a plot of the density in one dimension.

Fig. 10a shows the density from the first two-dimensional Riemann problem run with the third-order ADER-WENO code along with the multidimensional HLLC Riemann solver. As a counterfoil, Fig. 10b shows an entirely similar simulation but with the multidimensional HLL Riemann solver. There are 31 density contour levels between 0.5 and 1.7.

Fig. 11a shows the density from the second two-dimensional Riemann problem run with the third-order ADER-WENO code along with the multidimensional HLLC Riemann solver. As a counterfoil, Fig. 11b shows an entirely similar simulation but with the multidimensional HLL Riemann solver. There are 32 density contour levels between 0.15 and 1.7.

Fig. 12a shows the density from the double Mach reflection problem run with the third-order ADER-WENO code along with the multidimensional HLLC Riemann solver. Fig. 12b zooms in on the roll-up of the Mach stem. There are 31 density contour levels between 2.5 and 21.5. Figs.

12c and 12d show analogous information to Figs. 12a and 12b. However, Figs. 12c and 12d show the results from a third-order ADER-WENO code along with a one-dimensional HLLC Riemann solver run with CFL 0.475.

Fig. 13 shows the density from the forward facing step problem run with the third-order ADER-WENO code along with the multidimensional HLLC Riemann solver. There are 41 density contour levels between 0.5 and 4.5.

Fig. 14 shows the results from a blast wave problem. Fig. 14a shows the density as a vertical height with the mesh overlaid. Fig. 14b shows the angularly averaged density as a function of radius. Figs. 14c and 14d show the same for the pressure and velocity.

Fig. 15 shows the final result for the magnetized rotor problem, at a time of 0.25. Figs. 15a, 15b, 15c and 15d show the density, pressure, magnetic pressure and Mach number at the final time. The density has 17 contours in the range [1.5, 9.5]; the pressure has 44 contours in the range [0.1, 1.2]; the magnetic pressure has 33 contours in the range [0.05, 0.85]; the Mach number has 40 contours in the range [0, 2.0].

Figs. 16a, 16b, 16c and 16d show the final density, pressure, magnitude of the velocity and the magnitude of the magnetic field respectively for the Orzag-Tang problem. The density has 26 contours in the range [1, 6];the pressure has 30 contours in the range [0.4, 6.2]; the velocity magnitude has 35 contours in the range [0.0, 1.7]; the magnetic field magnitude has 45 contours in the range [0.0, 11.0].

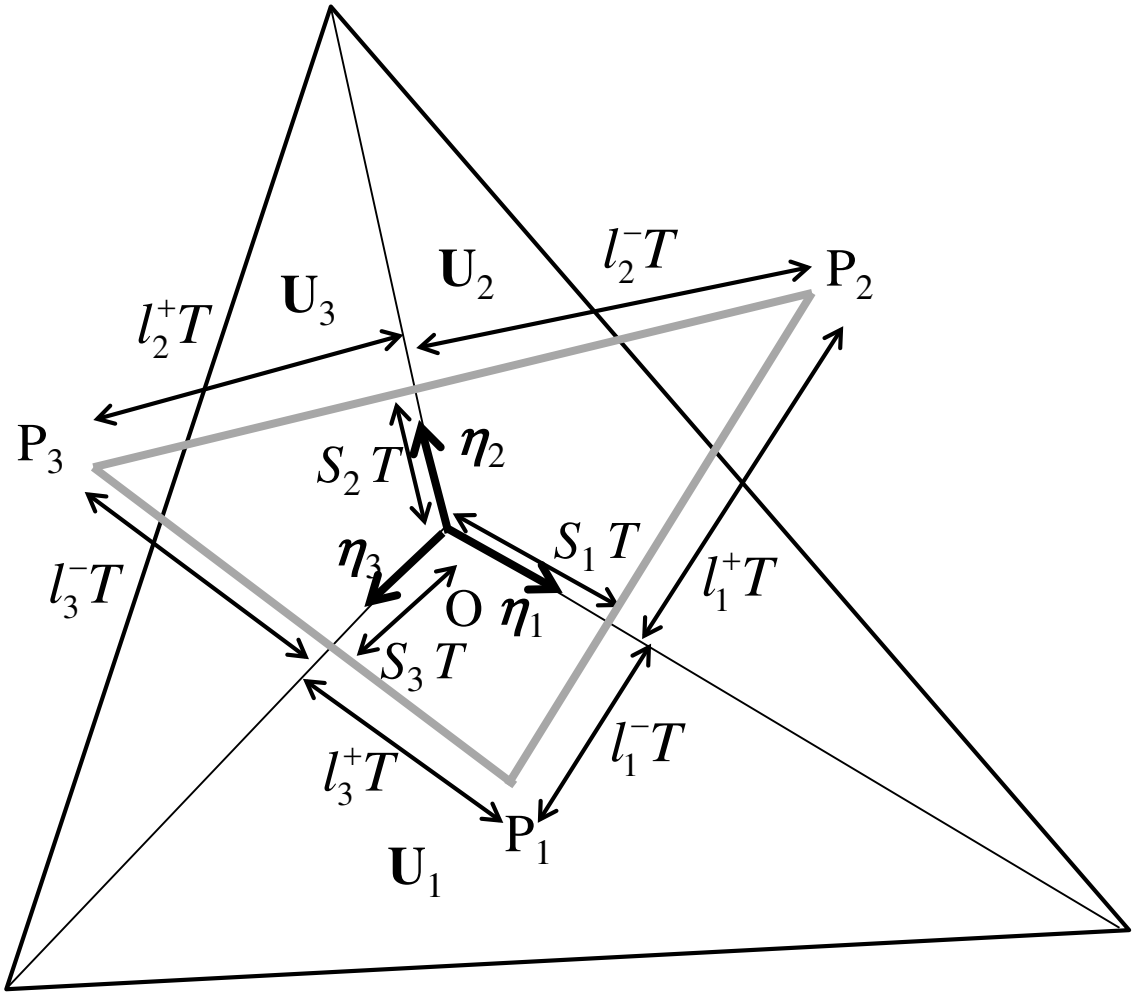


Fig. 1 shows the three states coming together at a vertex "O". The states are given by $\mathbf{U}_1$, $\mathbf{U}_2$ *and* $\mathbf{U}_3$. *They are separated by unit vectors* $\boldsymbol{\eta}_1$, $\boldsymbol{\eta}_2$ *and* $\boldsymbol{\eta}_3$ *with the counterclockwise orientation shown. The fastest waves propagating along these unit vectors do so with speeds* S_1, S_2 *and* S_3 *, so that their location at a time "T" is shown by the grey lines. The intersection of these grey lines forms a triangle and the strongly interacting state is contained within that triangle.*

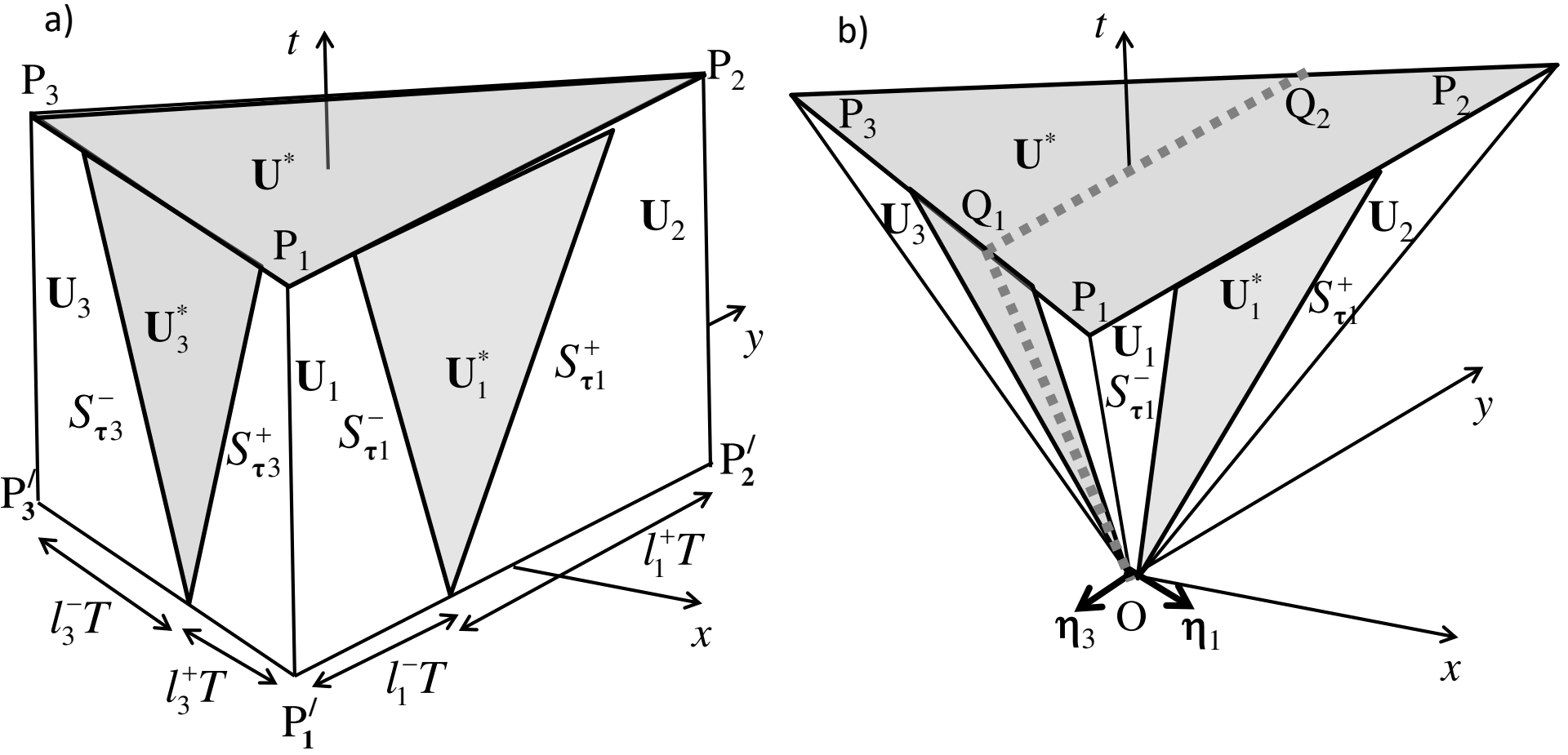


Fig. 2a shows the triangular prism in space and time that has the strongly interacting state **U*** *for its top surface. The side panels of Fig. 2a depict the one-dimensional HLL Riemann problems. Imagine now that the side panels of Fig. 2a are removed to expose the space-time domain covered by the strongly interacting state. Fig. 2b shows the self-similar extent of that strongly interacting state in space-time. The projection of the one-dimensional HLL Riemann problems are also shown on the side surfaces of Fig. 2b. The triangle* OQ_1Q_2 *shows the surface used for obtaining the x-flux.*

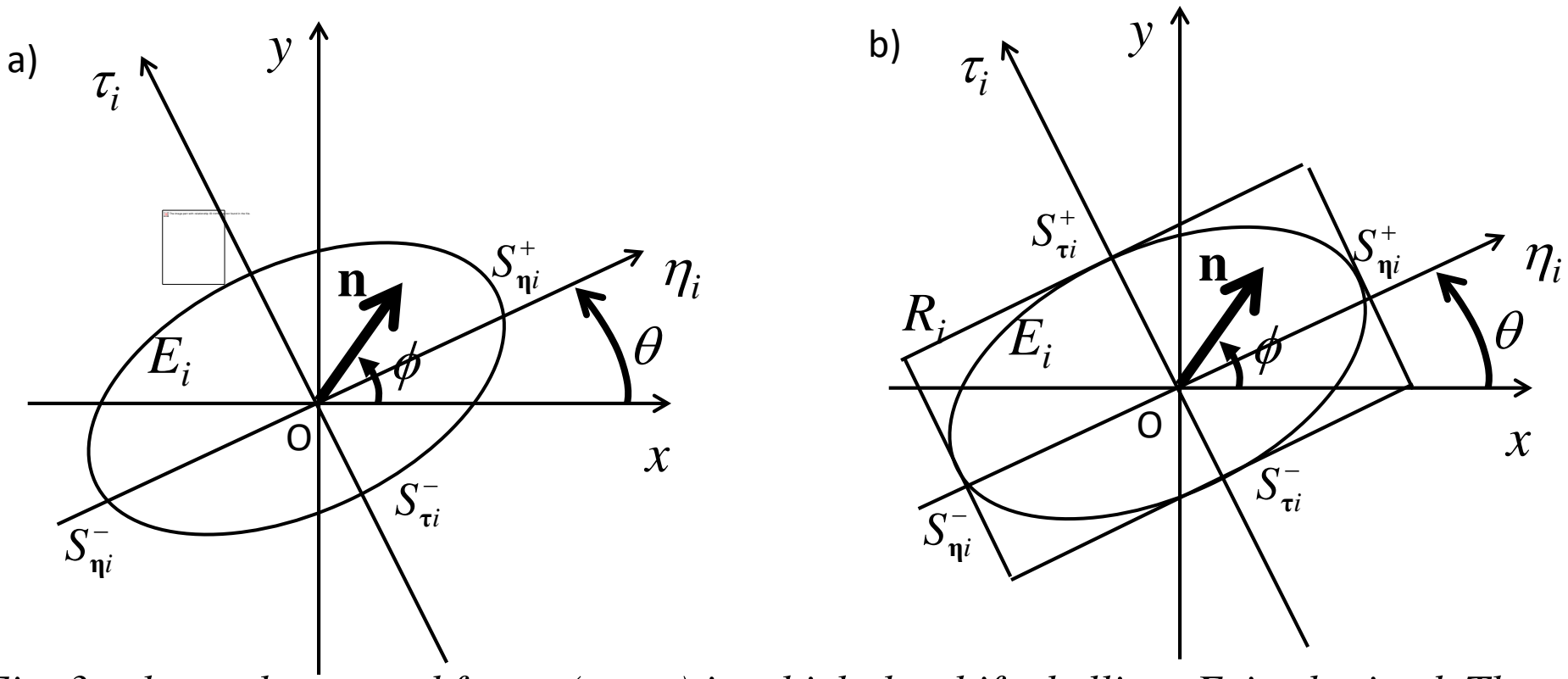


Fig. 3a shows the rotated frame (η_i , τ_i) in which the shifted ellipse E_i is obtained. The rotated frame makes an angle θ with the global (x,y) frame. Ellipse E_i represents the wave propagation in the rotated frame. The extent of the ellipse E_i in the rotated frame is also shown by the speeds "S". The unit vector in this frame along which we wish to find the maximal speed is shown as **n** *. It makes an angle ϕ with the global (x,y) frame. Fig. 3b shows how the ellipse E_i in Fig. 3a is replaced with a bounding rotated rectangle R_i . The rectangle R_i circumscribes the ellipse and will, therefore, be a good proxy for the maximal speeds that can be represented by the ellipse. The vertices of the rectangle R_i are indeed easily specified in the rotated (η_i , τ_i) frame. The projection of those vertices in the direction of the unit vector* **n** *gives the maximal speed contributed by R_i along* **n**.

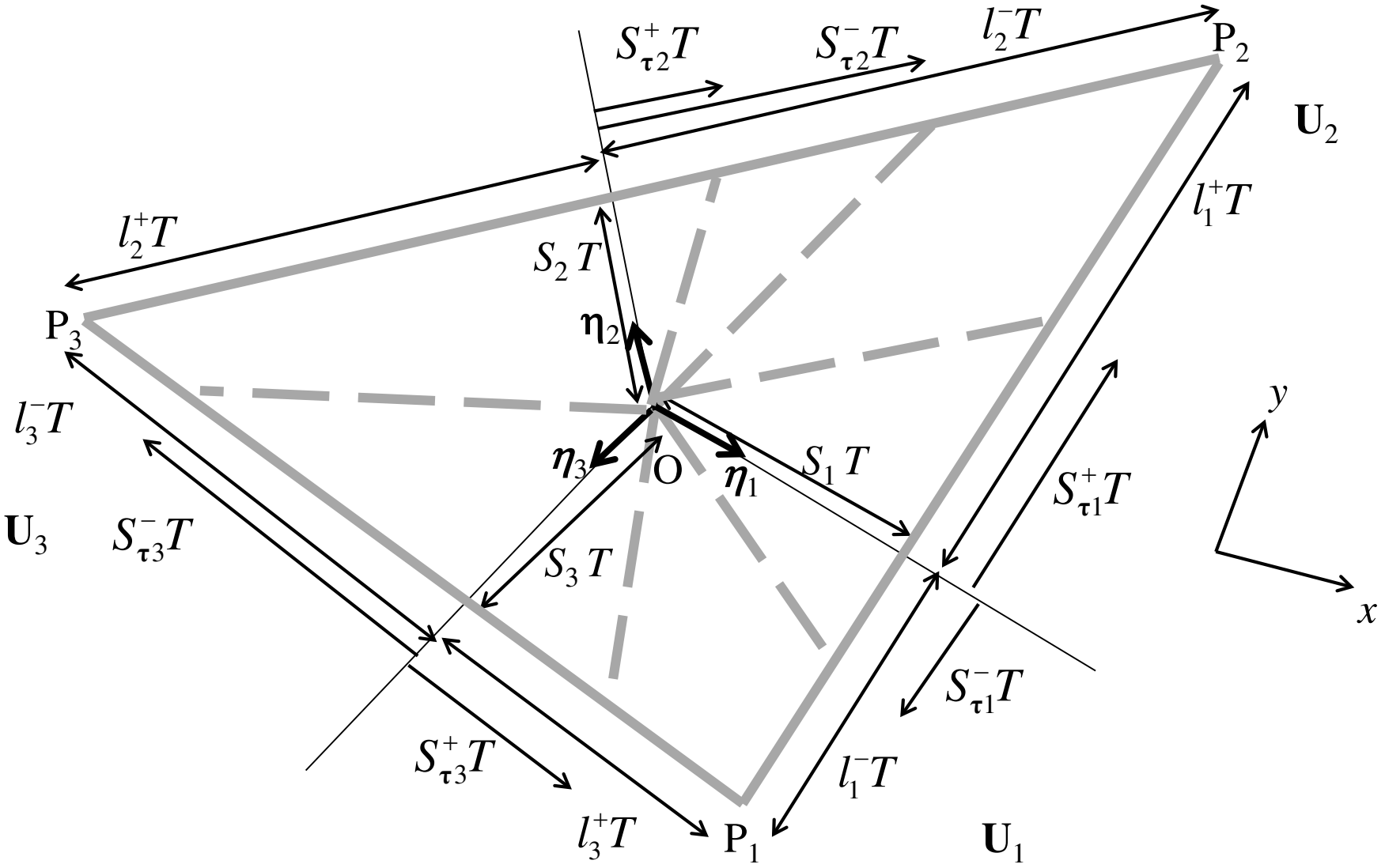


Fig. 4 shows a magnified view of the grey triangle from Fig. 1. The dashed lines demarcate the boundaries of the intermediate states associated with the one-dimensional HLL Riemann problems. The dashed lines are two-dimensional projections of the solid lines within the side panels of Fig. 2b. As depicted, the 1D Riemann problem is subsonic in faces 1 and 3 and supersonic in face 2. The orientation of the x- and y-axes is shown.

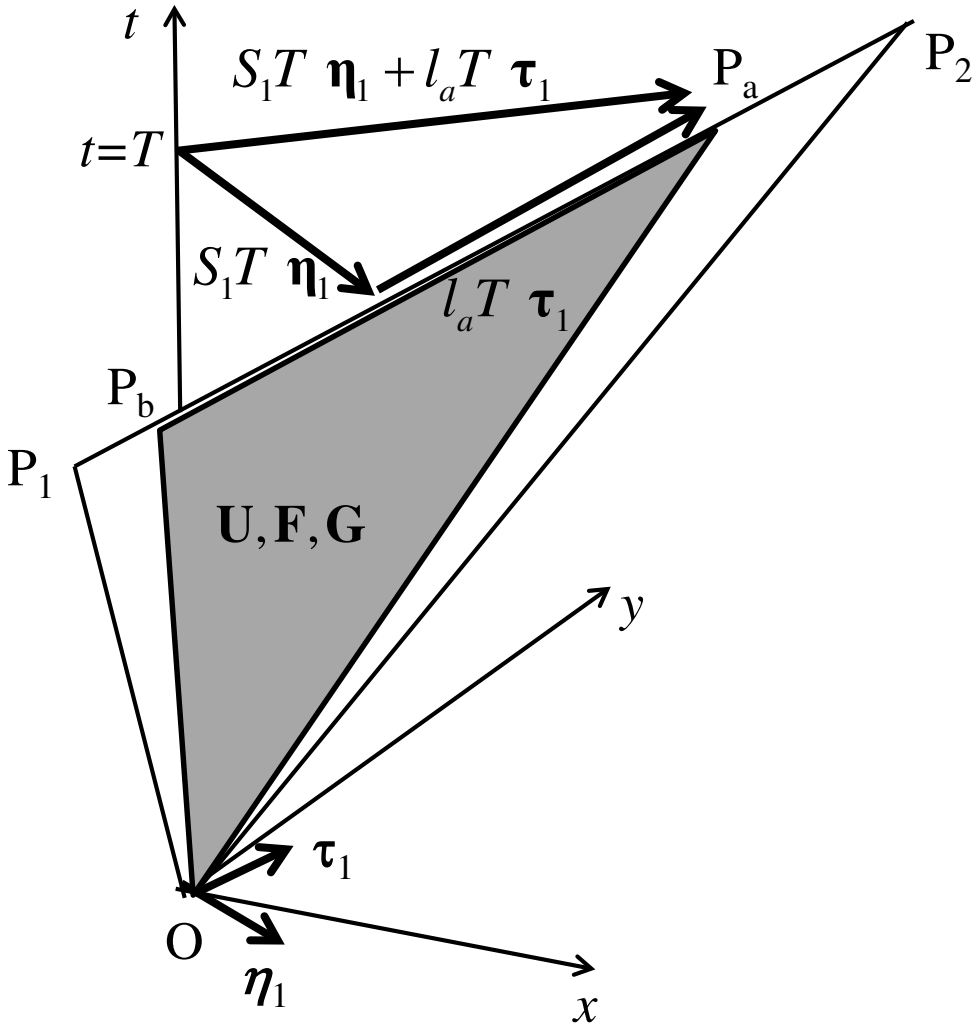
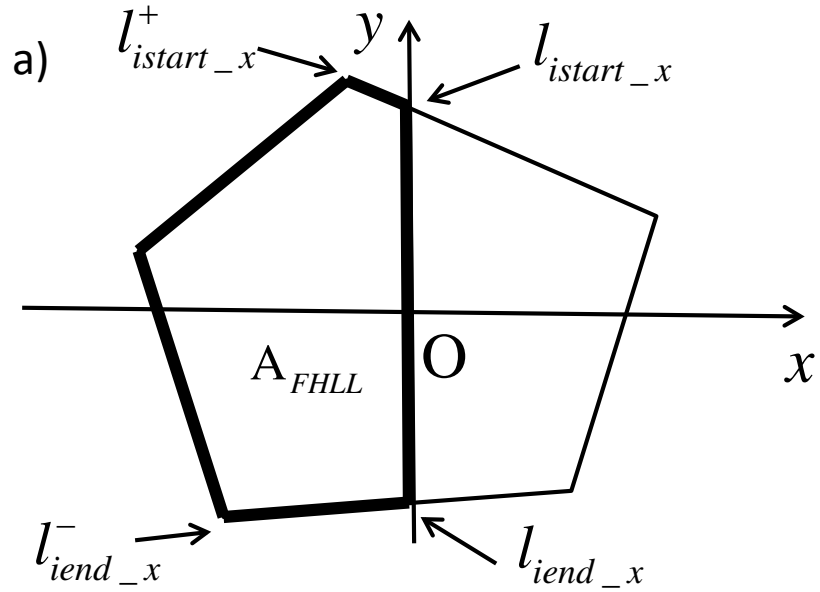
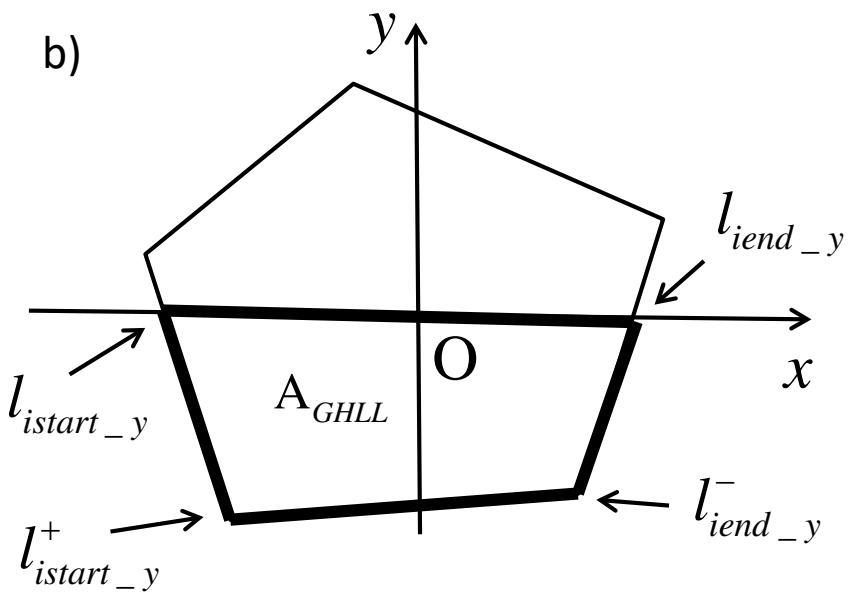
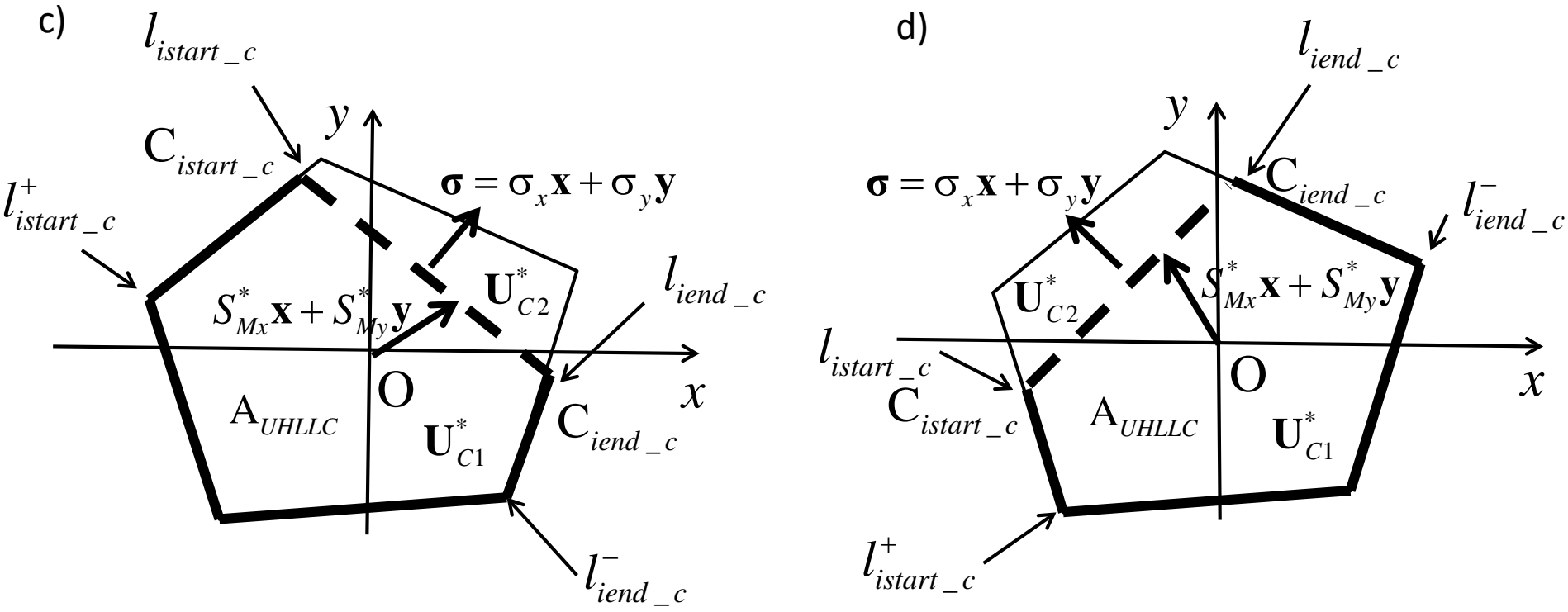


Fig. 5 shows an expanded version of triangle ΔOP_1P_2 *from Fig. 2b. It helps in illustrating the procedure for evaluating the area integral over a sub-triangle* $\Delta\ OP_aP_b$. *The state* **U** *and fluxes* **F** *and* **G** *are constant on this sub-triangle. The text shows how the facial integrals over sub-triangles are easily evaluated in various circumstances.*





Figs. 6a and 6b show the domains used for evaluating the x- and y-fluxes for the multidimensional HLL Riemann solver. A five-sided wave model is shown.



Figs. 6c and 6d show different propagation directions of the contact discontinuity for the multidimensional HLLC Riemann solver. The contact discontinuity is shown by the thick dashed line. Its orientation is given by the unit normal $\boldsymbol{\sigma}$.

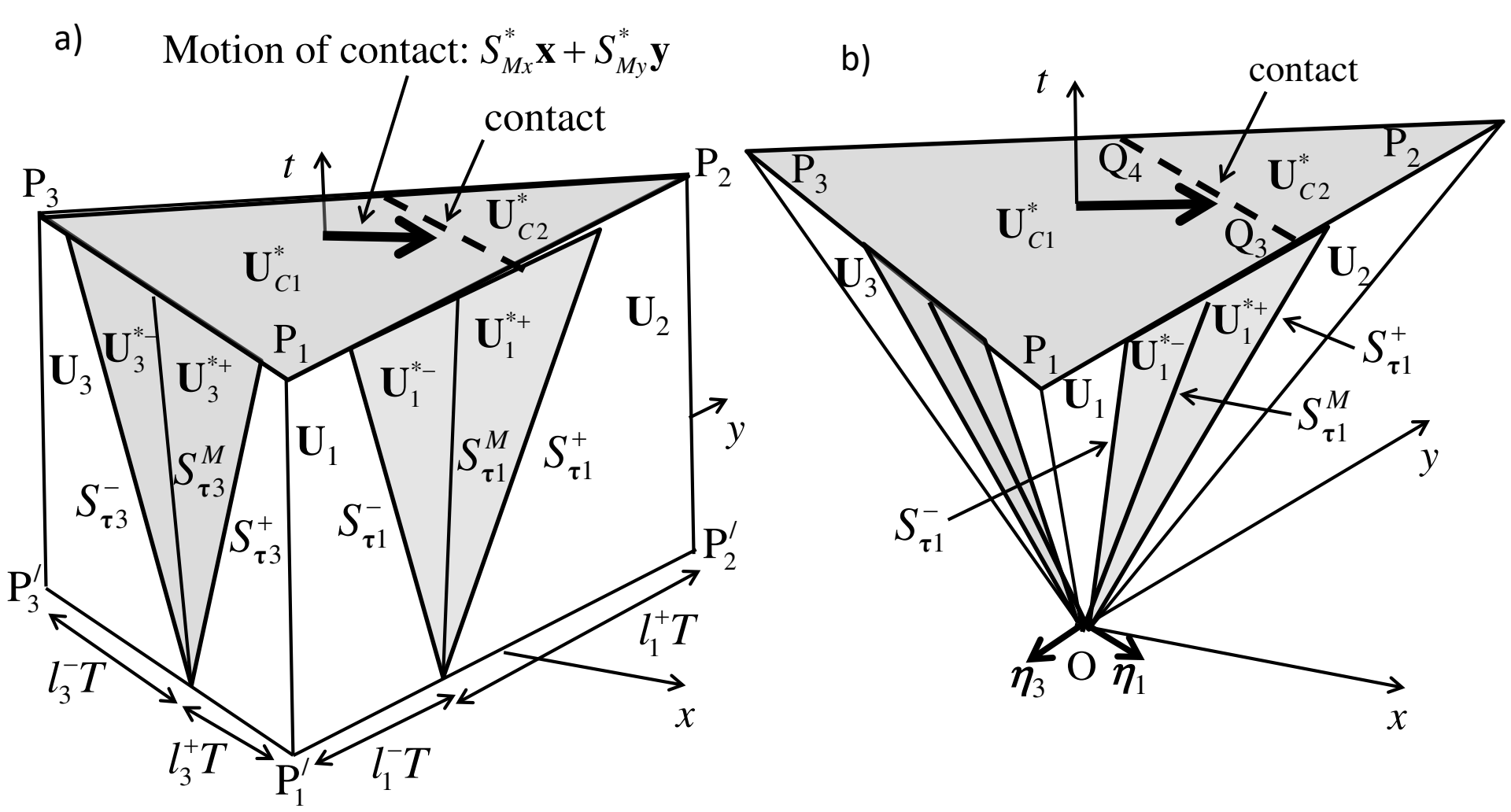
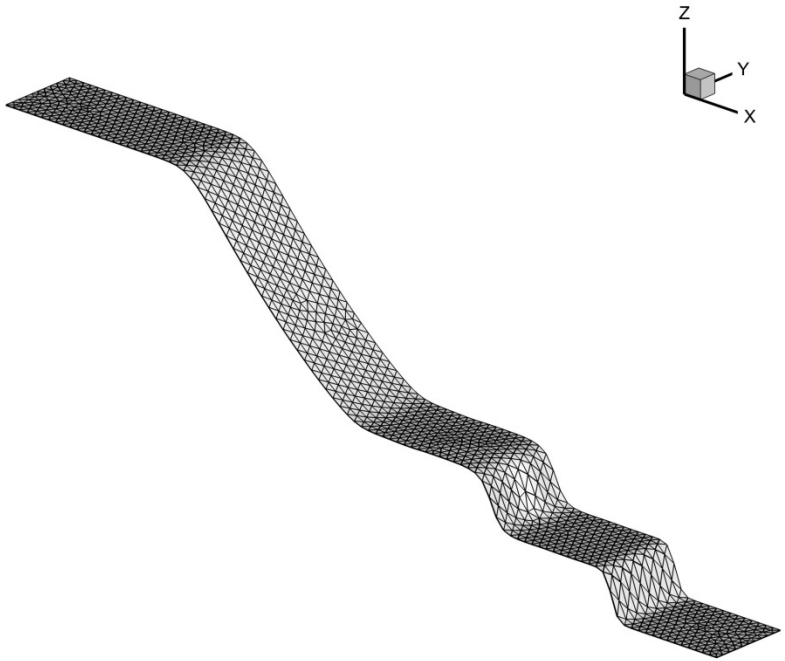
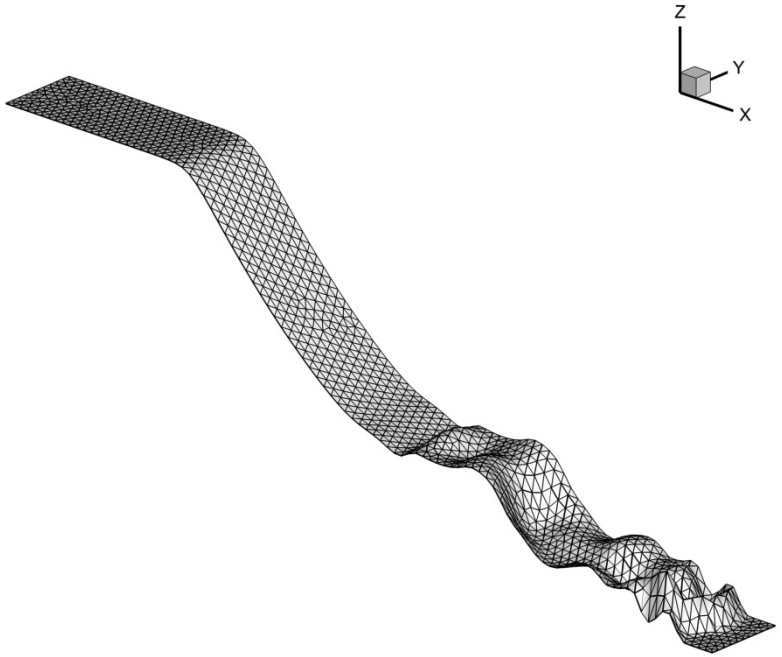


Fig. 7a shows the triangular prism in space and time that is used for obtaining the strongly interacting states for the multidimensional HLLC Riemann solver. The contact discontinuity, shown as a thick dashed line, separates the two strongly interacting states. Fig. 7b can be thought of as removing the side panels of Fig. 7a. The strongly-interacting states form an inverted triangular pyramid in space-time. The waves and states in Fig. 7b are only labeled in the face corresponding to η_1 .

a)



b)



c)

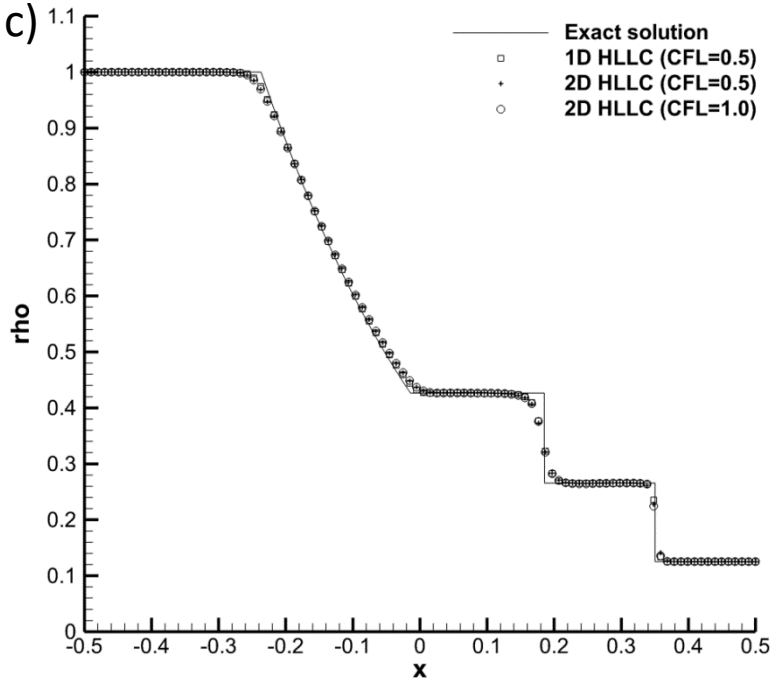
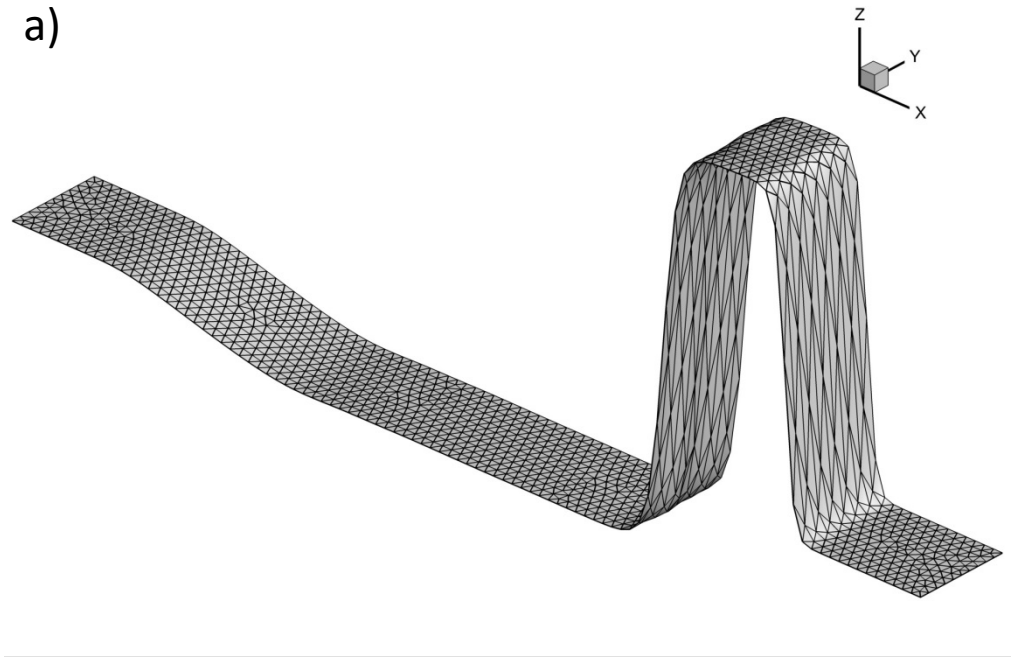


Fig. 8 shows results from the Sod shock tube problem done on a 2D unstructured mesh. Fig. 8a shows the density and the mesh for a simulation that was run with the multidimensional HLLC Riemann solver with a CFL of 1; Fig. 8b shows the same when the 1D HLLC Riemann solver was used with the same CFL. Fig. 8c shows a plot of the density in one dimension for various simulations. Fig. 8b shows us that the simulation goes unstable when a 1d Riemann solver is used with a CFL of unity.



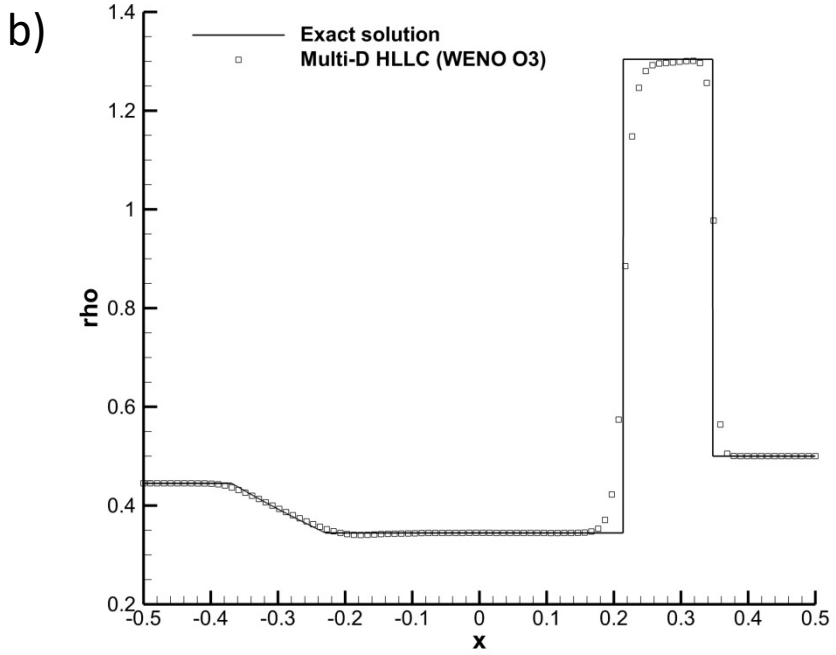


Fig. 9 shows results from the Lax shock tube problem done on a 2D unstructured mesh. Fig. 9a shows the density and the mesh for a simulation that was run with the multidimensional HLLC Riemann solver with a CFL of 1. Fig. 9b shows a plot of the density in one dimension.

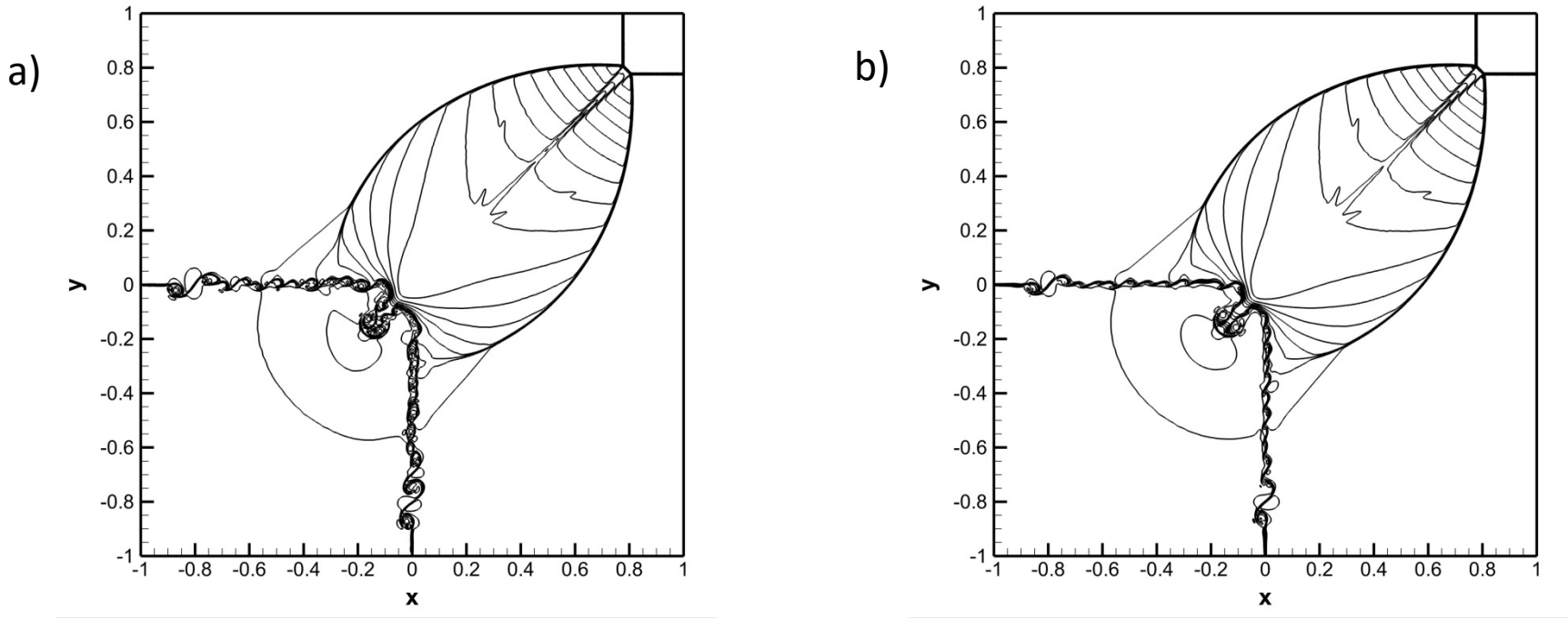


Fig. 10a shows the density from the first two-dimensional Riemann problem run with the third-order ADER-WENO code along with the multidimensional HLLC Riemann solver. As a counterfoil, Fig. 10b shows an entirely similar simulation but with the multidimensional HLL Riemann solver. There are 31 density contour levels between 0.5 and 1.7.

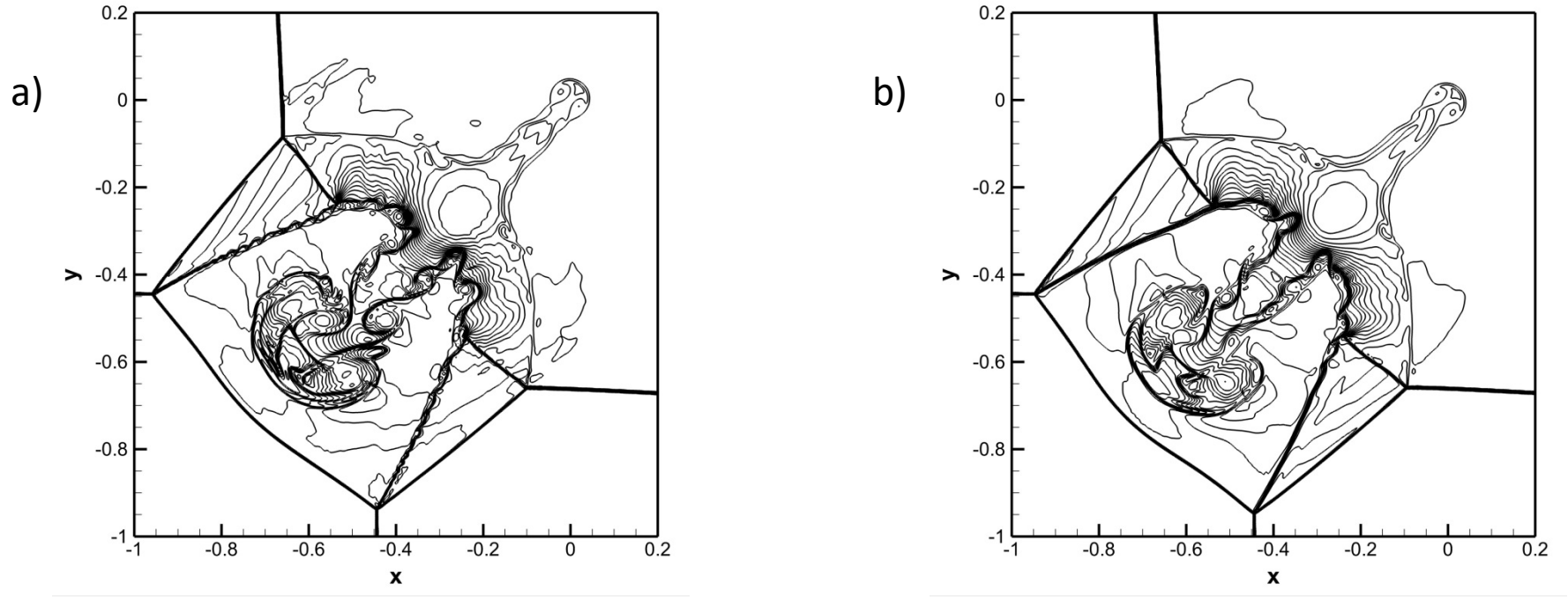
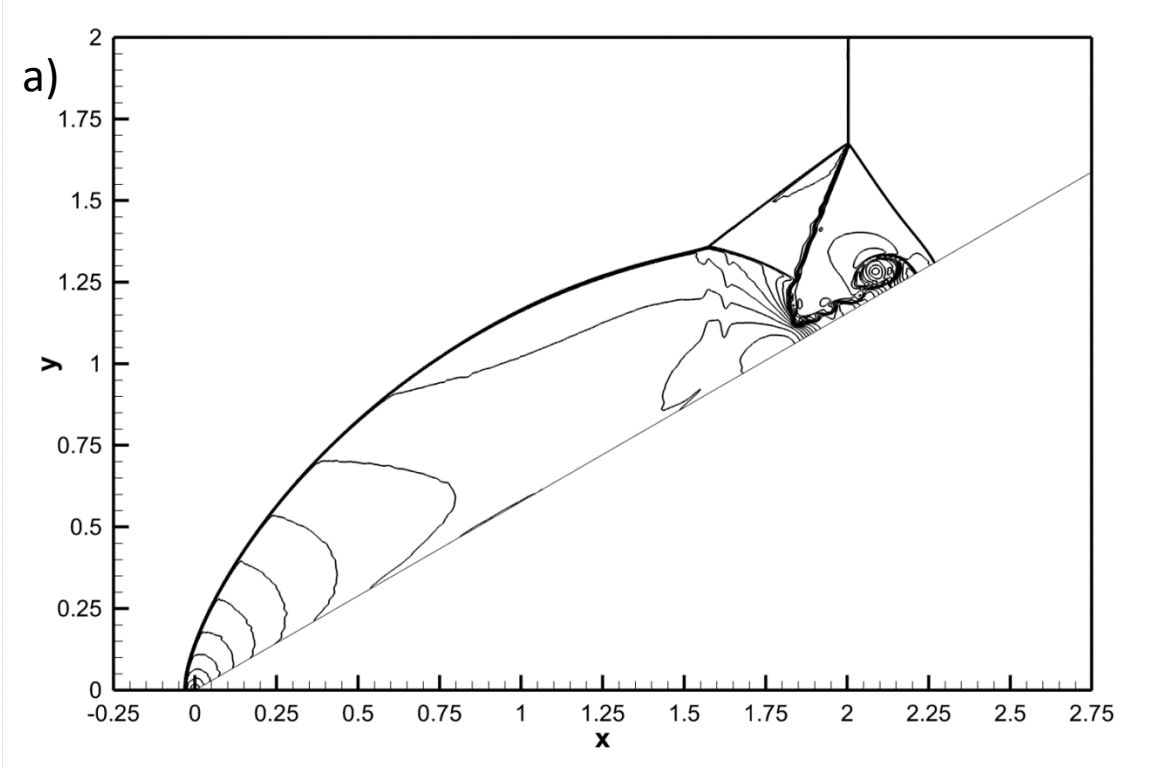
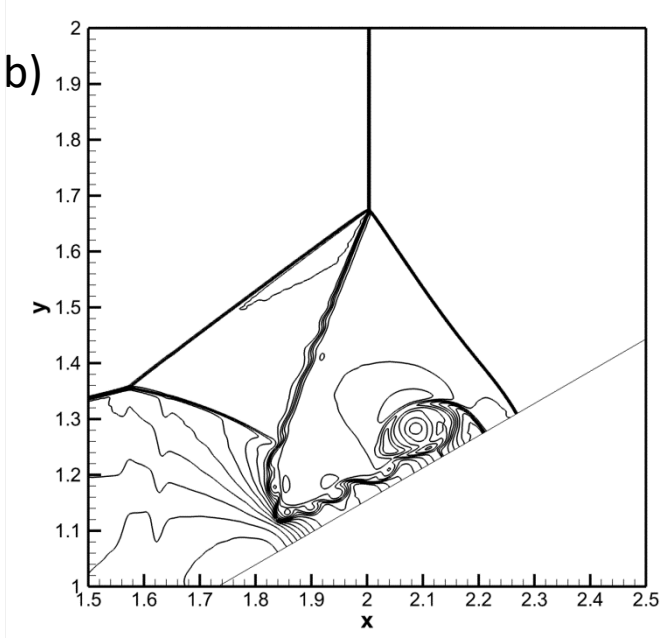
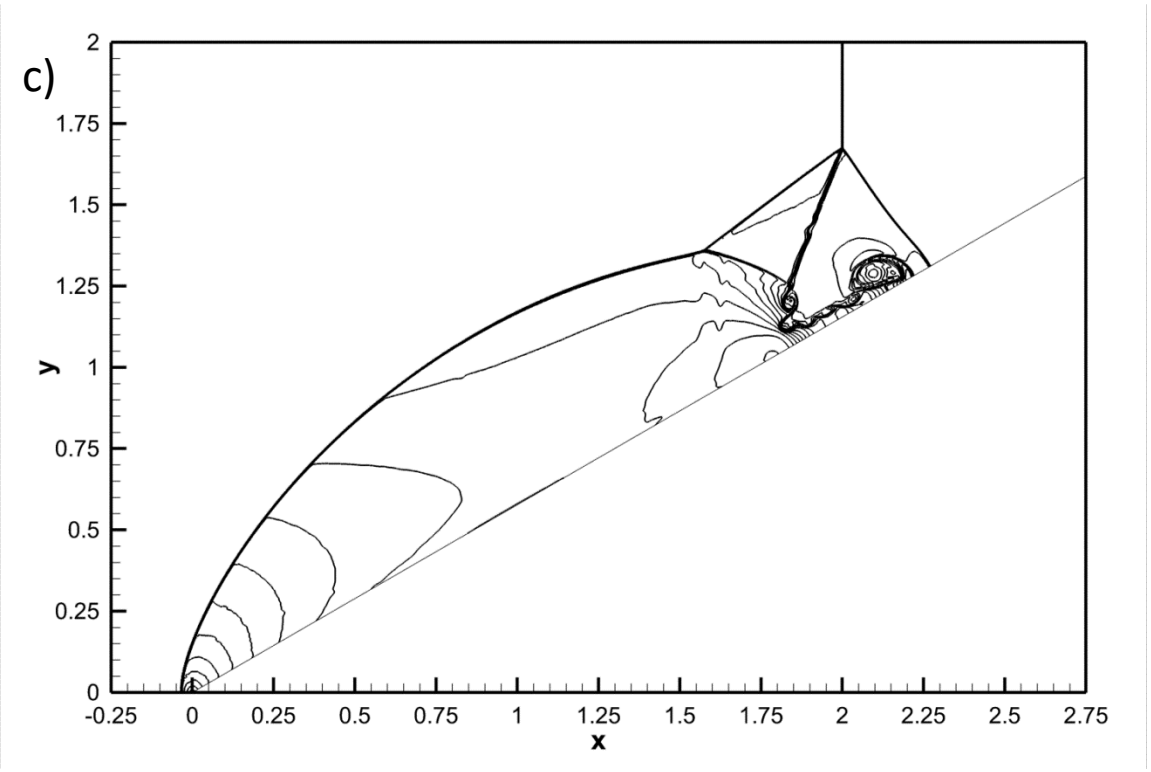


Fig. 11a shows the density from the second two-dimensional Riemann problem run with the third-order ADER-WENO code along with the multidimensional HLLC Riemann solver. As a counterfoil, Fig. 11b shows an entirely similar simulation but with the multidimensional HLL Riemann solver. There are 32 density contour levels between 0.15 and 1.7.


a)
2
1.75
1.5
1.25
1
0.75
0.5
0.25
0
y
-0.25
0
0.25
0.5
0.75
1
1.25
1.5
1.75
2
2.25
2.5
2.75
x


b)
2
1.9
1.8
1.7
1.6
1.5
1.4
1.3
1.2
1.1
1
y
1.5
1.6
1.7
1.8
1.9
2
2.1
2.2
2.3
2.4
2.5
x



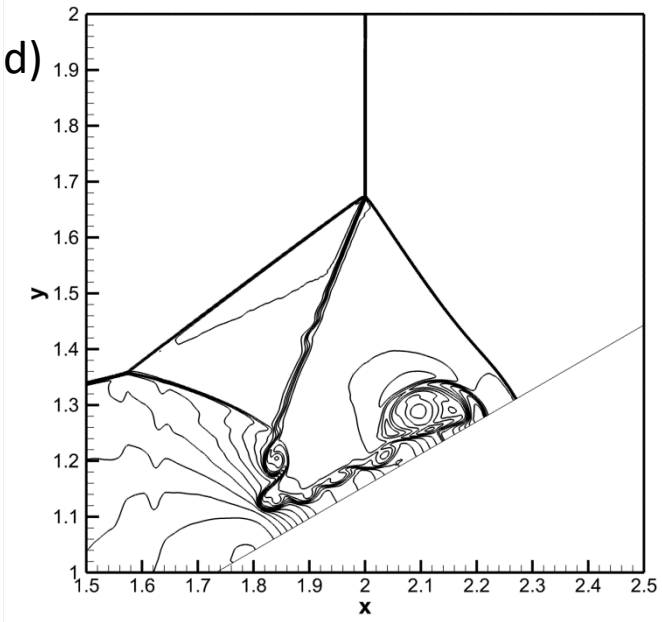


Fig. 12a shows the density from the double Mach reflection problem run with the third-order ADER-WENO code along with the multidimensional HLLC Riemann solver. Fig. 12b zooms in on the roll-up of the Mach stem. There are 31 density contour levels between 2.5 and 21.5. Figs. 12c and 12d show analogous information to Figs. 12a and 12b. However, Figs. 12c and 12d show the results from a third-order ADER-WENO code along with a one-dimensional HLLC Riemann solver run with CFL 0.475.

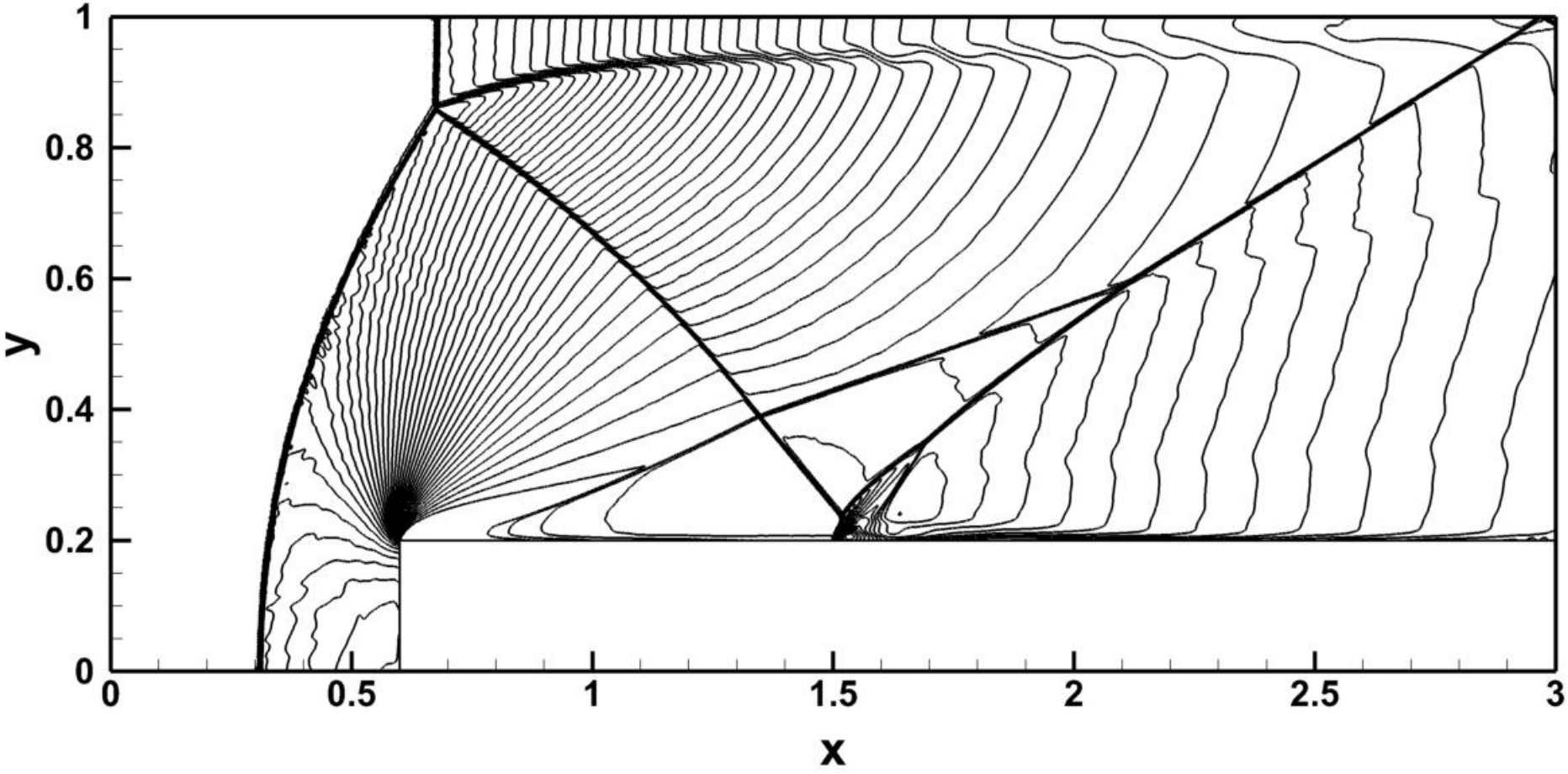


Fig. 13 shows the density from the forward facing step problem run with the third-order ADER-WENO code along with the multidimensional HLLC Riemann solver. There are 41 density contour levels between 0.5 and 4.5.

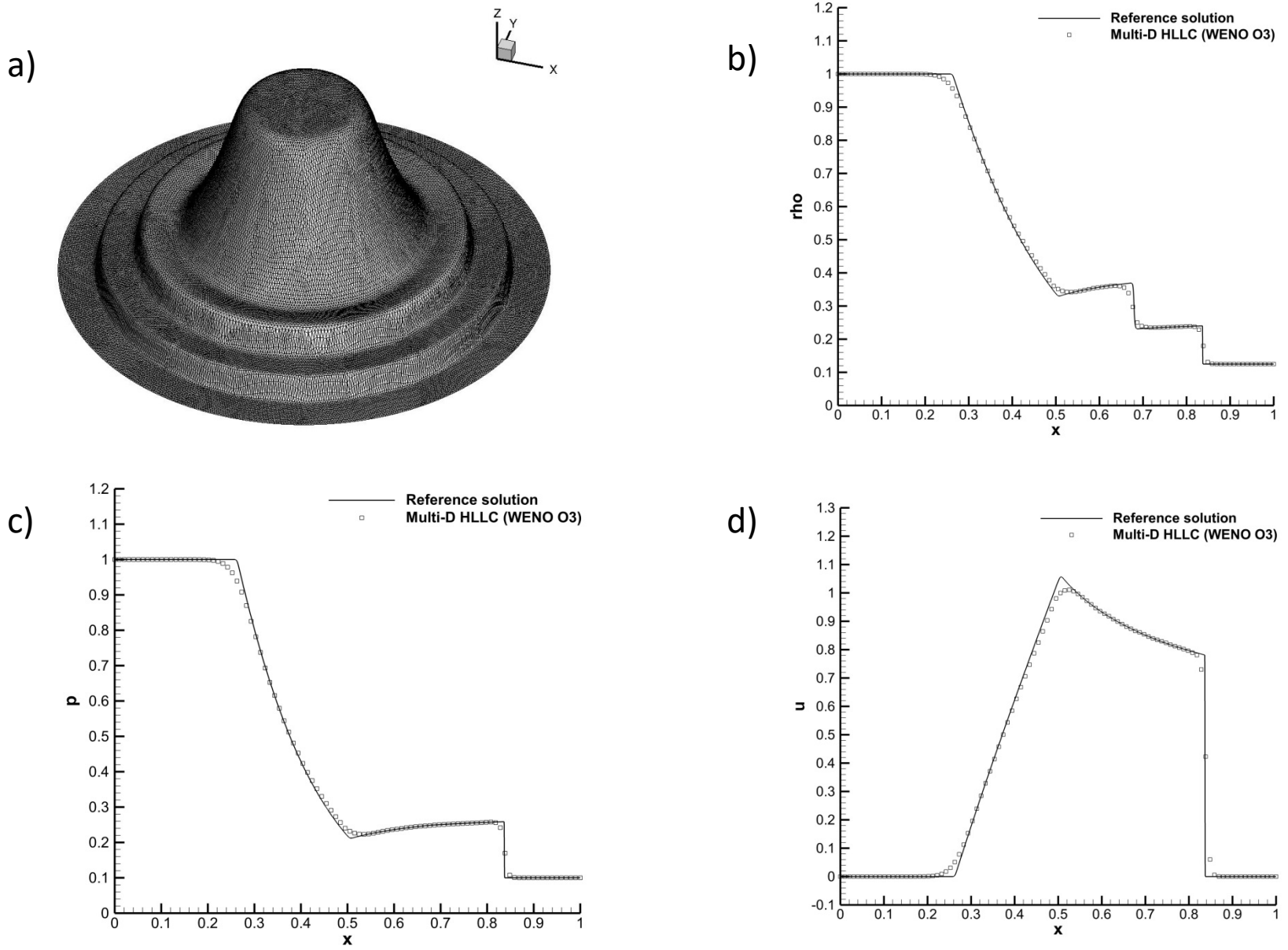


Fig. 14 shows the results from a blast wave problem. Fig. 14a shows the density as a vertical height with the mesh overlaid. Fig. 14b shows the angularly averaged density as a function of radius. Figs. 14c and 14d show the same for the pressure and velocity.

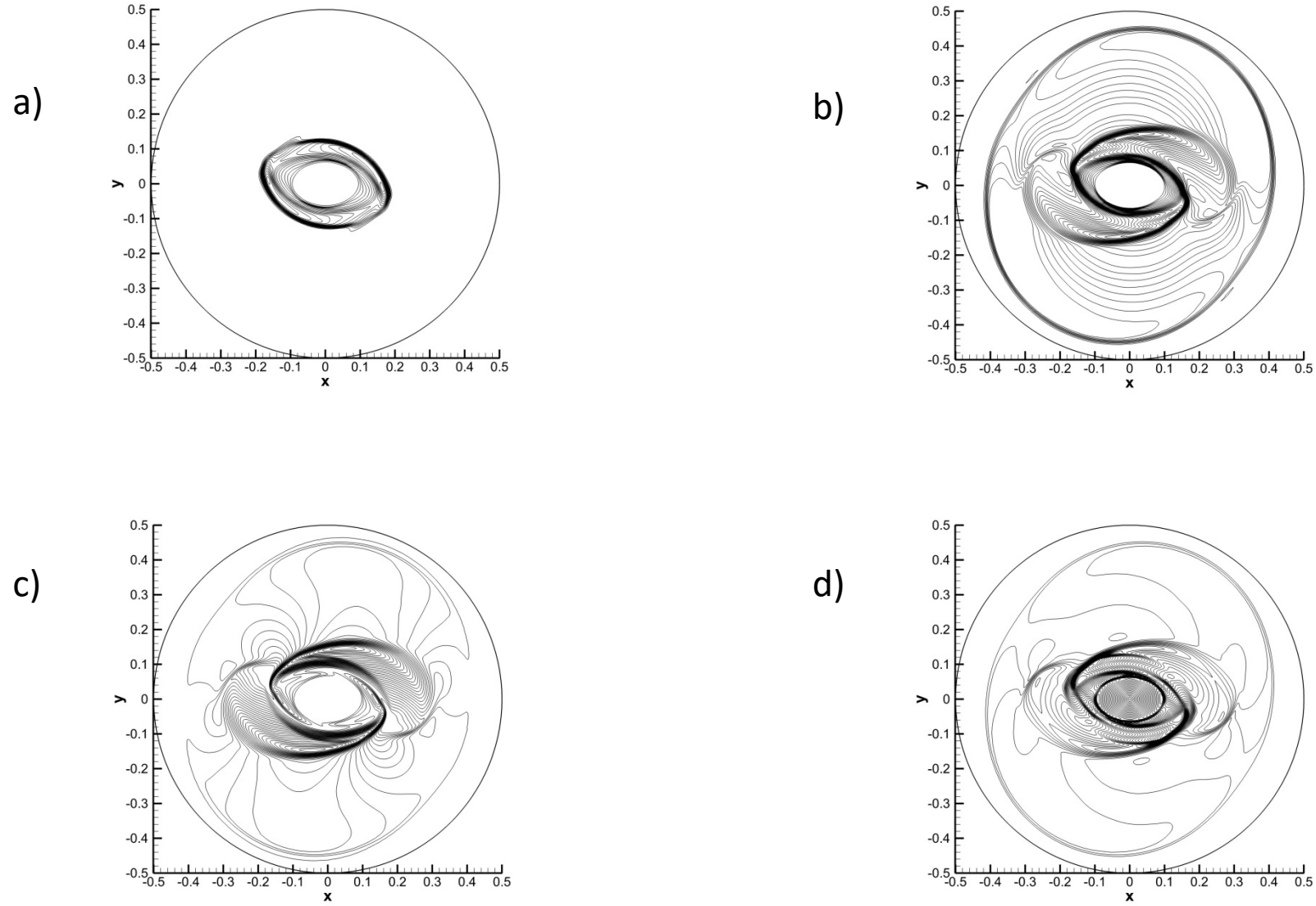
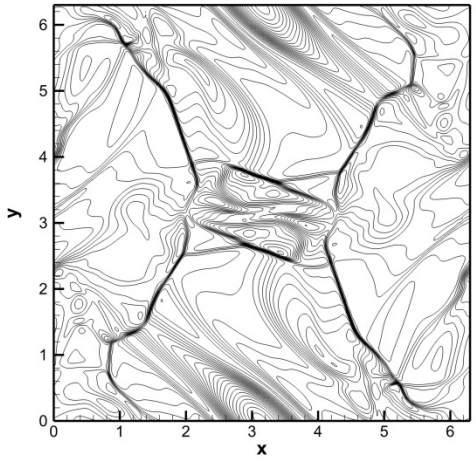
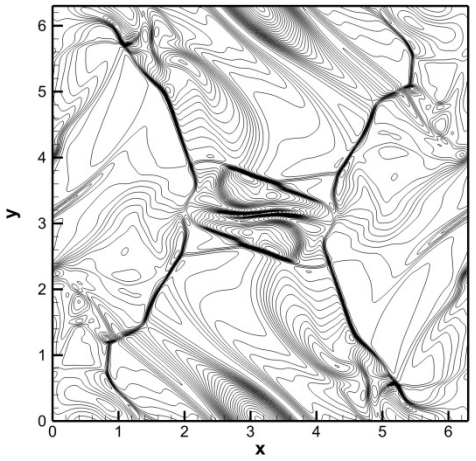


Fig. 15 shows the final result for the magnetized rotor problem, at a time of 0.25. Figs. 15a, 15b, 15c and 15d show the density, pressure, magnetic pressure and Mach number at the final time. The density has 17 contours in the range [1.5, 9.5]; the pressure has 44 contours in the range [0.1, 1.2]; the magnetic pressure has 33 contours in the range [0.05, 0.85]; the Mach number has 40 contours in the range [0, 2.0].

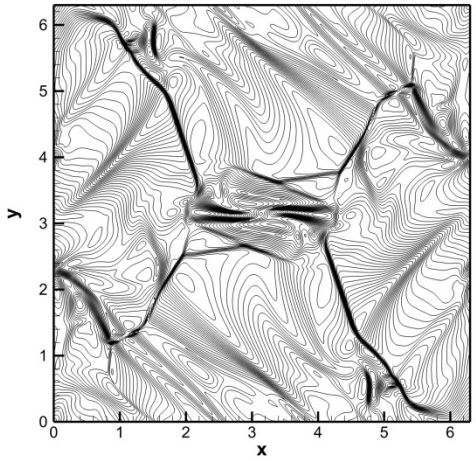
a)



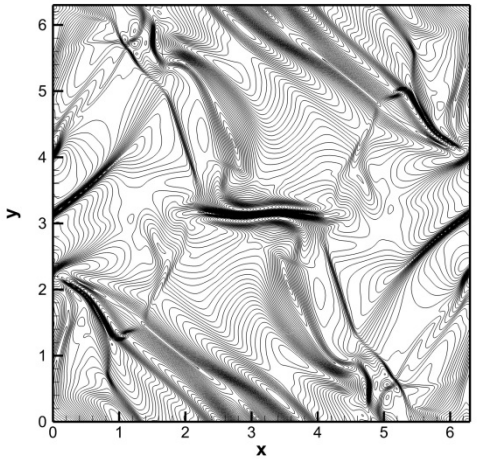
b)



c)



d)



Figs. 16a, 16b, 16c and 16d show the final density, pressure, magnitude of the velocity and the magnitude of the magnetic field respectively for the Orzag-Tang problem. The density has 26 contours in the range [1, 6];the pressure has 30 contours in the range [0.4, 6.2]; the velocity magnitude has 35 contours in the range [0.0, 1.7]; the magnetic field magnitude has 45 contours in the range [0.0, 11.0].